EN RU PROGRAM SYSTEMS: THEORY AND APPLICATIONS ISSN 2079-3316
Review Article HARDWARE AND SOFTWARE FOR DISTRIBUTED AND SUPERCOMPUTER SYSTEMS

UDC 004.051+004.272+004.318+004.382+004.8+004.9
10.25209/2079-3316-2025-16-5-43-514

Around the Conditional 4th Generation of Modern Server Processors AMD and Intel: their Microarchitecture and the Performance of the Corresponding Computing Systems

Mikhail Borisovich **Kuzminsky**✉
Zelinsky Institute of Organic Chemistry of RAS, Moscow, Russia
✉*kus@free.net*

Abstract. This review focuses on the microarchitecture and performance of Intel Xeon processors—4th-generation Scalable processors (with the Sapphire Rapids-SP microarchitecture, hereafter Xeon SPR), 5th-generation (Emerald Rapids-SP, hereafter Xeon EMR), and various classes of AMD EPYC processors based on the Zen 4 architecture—as well as computing systems based on them. Data is analyzed for Xeon SPR models (and Xeon SPR with HBM memory, i.e., Xeon Max), Xeon EMR, and AMD EPYC 9004 processors (although brief data on the EPYC 8004 and 4004 is also provided).

These processors are classified in this review as belonging to the 4th generation of Xeon and EPYC processors. Comparisons are also made with 3rd-generation Xeon Scalable Processors (Ice Lake-SP, hereafter referred to as Xeon ICL), Cooper Lake-SP, AMD EPYC with the Zen 3 (Milan) architecture, and occasionally with ARM processors and GPUs.

The software development kits (SDKs) for 4th-generation processors, which are crucial for the achieved performance, are briefly discussed. Due to the use of chiplets or HBM memory in the AMD and Intel processors under consideration, special attention is paid to the supported NUMA variants.

Hardware support for security features for virtualization tasks, which are now often used in high-performance computing (HPC), is also analyzed.

The performance data in the review covers a wide range of application areas typical for servers with these processors, but the primary focus is on HPC and, to a lesser extent, AI workloads.

The processors in question are analyzed from the perspective of building homogeneous or GPU-enabled heterogeneous servers and computing systems based on them (clusters and supercomputers).

Initial information on the latest Intel Xeon 6 Granite Rapids and AMD EPYC Zen 5 Turin processors, including initial performance data, is also analyzed.

General conclusions are drawn about the status and emerging development trends of these x86 processors. (*Linked article texts in English and in Russian*).

Key words and phrases: x86, Zen 4, Genoa, Bergamo, Zen 5,Turin, Sapphire Rapids, Xeon Max, Emerald Rapids, Xeon 6, Granite Rapids, microarchitecture, performance, HPC, AI, supercomputers

2020 *Mathematics Subject Classification*: 65Y05; 68M20

For citation: Mikhail B. Kuzminsky. *Around the Conditional 4th Generation of Modern Server Processors AMD and Intel: their Microarchitecture and the Performance of the Corresponding Computing Systems*. Program Systems: Theory and Applications, 2025, **16**:5(68), pp. 43–514. (*In English, in Russian*).
https://psta.psiras.ru/read/psta2025_5_43-514.pdf

Introduction

Modern x86-64 processors from Intel and AMD remain the most common server processors in the world for a wide range of applications. In the field of supercomputers, this is easy to see from the TOP500 list [1]. This is also the case for all areas of HPC and AI, although for such tasks, x86-64 processors are also very actively used in heterogeneous servers together with GPUs [2]. And among those areas in which GPUs are still not used, for example, nuclear safety tasks are indicated [3].

To clarify the comparative performance indicators of such processors and GPUs, it is useful to immediately provide an illustration. For example, in [4] it is shown that on a dual-processor (2P-) system with an Intel Xeon Platinum 8480+, when multiplying square matrices of different sizes, the performance achieved was very close to that obtained on a single Nvidia A100 GPU without using tensor cores, although the power consumption when working with the GPU was much lower.

But modern supercomputers are often built on homogeneous nodes without accelerators. In the November 2023 TOP500 list [5], 22% of the 50 most powerful supercomputers were built on homogeneous servers with x86-64 processors from Intel and AMD, including the latest 4th generation processors considered in this review. Although in the June 2024 TOP500 list, this share decreased to 14% due to the appearance of supercomputers with the latest GPUs, new supercomputers with homogeneous nodes are also appearing in the TOP500.

The review analyzes only x86-64 server processors, which will be referred to as x86 for brevity. And the generations of x86 processors hereinafter refers to the generations of AMD EPYC Zen architectures and scalable Xeon processors.

AMD EPYC Zen 4 and 4th and 5th generation Xeon Scalable processors (Xeon SPR and Xeon EMR), the basic subject of consideration in this review, are referred to here as the conditional 4th generation of x86.

The Chinese SW39000 RISC processors [6] have become the world leaders in peak performance (14 TFLOPS for FP64), but they are mainly aimed at the Sunway supercomputer.

The use of ARM server processors is expanding worldwide, but they have not reached the level of a clear performance advantage over x86 (this was also the case immediately after the appearance of the Fujitsu A64FX [7]). ARM mainly competes with x86 in terms of energy efficiency (the ratio of performance to energy consumption) and in the use of cloud technology (see, for example, [8–10]), although ARM has also shown success in the cost/performance ratio (see, for example, [11]).

The use of new server ARM processors in HPC could begin to expand due to the use of high-performance ARM processors based on the Neoverse V2 architecture, but apart from the 72-core Nvidia Grace in heterogeneous systems with Nvidia H100 and H200 GPUs (see, for example, data [2, 12]), at the time of writing, only the 96-core Amazon AWS Graviton 4 is known, and data demonstrating their significant successees in performance compared to x86 are practically absent.

An interesting comparison of the ARM Neoverse V2 microarchitecture in the Nvidia Grace Superchip and the x86 processors considered in the review is given in [13], where the competition between Nvidia, Intel and AMD in the race for the best processor is noted. However, the comparison of the 4th generation Intel and AMD processors with ARM is interesting in practical terms based on real performance, and not on the microarchitecture itself, and the data, including from [13], indicate the clear competitiveness of Grace mainly for servers with GPUs, where higher memory bandwidth is important for Nvidia Grace — due to the use of more expensive LPDDR5X memory (if we do not take into account the Xeon Max with even more expensive HBM). . The task of optimally selecting a processor for a specific application may also be important [14].

The performance leadership of x86 processors relative to other processors have no bearing on the advantages or disadvantages of CISC over RISC, since x86 hardware has long translated its instructions from x86 ISA into RISC micro-operations, which are then executed [15, 16]. Of course, this fact itself can be seen as a confirmation of the advantage of RISC, as can the recent APX extension of Intel's ISA [17], which will use the larger number of general-purpose registers and instructions that work with three registers, which are more typical of RISC.

But in reality, today a lot of hardware is required in each core for out-of-order (OoO) instruction processing with the required hardware blocks for scheduling the execution of subsequent commands on free execution units, for branch prediction and other components of the front-end part of modern superscalar processors, which give each processor core the ability to simultaneously execute the maximum possible number of commands as soon as execution units are available. And this takes more resources than required due to the differences between CISC and RISC.

The discussions on RISC vs CISC, which appeared a long time ago, became more active during the appearance of ARM [18–20], and continue to this day on the Internet. The actual "negligibility" of the advantages of RISC or CISC has been discussed for a long time [20]. In the author's opinion, for performance maximization tasks, the current more valid point of view is that the differences between RISC and CISC architectures are negligible

compared to the complex front-end implementations of the corresponding superscalar microprocessors used today (see, for example, [21]). Differences in the execution units of different processor architectures may prove more important.

And the advantages of modern x86 models today are related to their implementation at the microarchitecture level, the high level of the semiconductor technology used in them and the mass volume of deliveries, which provide financial benefits. And the integrally main thing that, from the point of view of the front-end part of the core, for example, AMD is focused on is the growth of the IPC (Instructions Per Clock) value. In AMD reports (for example, [22]) or modern AMD documents on Zen 4 (for example, [23]), the front-end is generally given rather little attention, since changes in the executive devices, higher levels of the memory hierarchy, and interconnections of cores and processors are more important. In the review, the front-end part of the x86 processors will also be considered relatively briefly.

Perhaps the most striking indicator of modern server processor families is the large number of cores used there. In Xeon SPR there are at least 8. In Xeon EMR there are at least 16, as also in EPYC Zen 4 9004 series. It is clear that there are tasks and applications that do not require even this number of cores. In addition to the usual ability to simultaneously run several applications within one OS, modern servers widely use virtualization capabilities, for which special hardware improvements are used in processors. Virtualization and cloud technologies have begun to be actively used for HPC tasks. And according to [24], at least 48 supercomputers occupying places 1 to 172 in the TOP500 (June 2022) use containers.

This review is focused as broadly as possible on HPC (including supercomputer level), less on AI, and even less on virtualization and cloud technologies.

The performance advantages of AMD EPYC in the field of virtualization and cloud technologies (for mass application not in the field of HPC and AI), including over Intel Xeon, are associated with a large number of cores available in processors and were formed five years ago. In general, there is a lot of data demonstrating the performance advantages of EPYC Zen 2 and Zen 3. We will indicate here only one example for HPC -article in the field of CFD [25].

A lot of relevant data for the widely used application areas of these server processors can be found, for example, in the results of various OSG (Open System Group) SPEC benchmarks, including SPEC CPU 2017 [26], or in the OpenBenchmarking [27] data for the Phoronix Test Suite (PTS) [28] benchmarks. The advantages of the above-mentioned EPYC series in this data can be seen, for example, by comparing the performance of the older models of these AMD processor generations with similar Xeon models at the corresponding times, or taking into account the cost/performance ratio.

This is also evident from a number of relevant publications on the Phoronix website (see, for example, [29–31]). Perhaps, possible exceptions mainly relate to cases with active use of AVX-512. A small information with performance comparisons with 3rd generation Xeon (Xeon ICL and Xeon Cooper Lake-SP) and EPYC Zen 3 (Milan) is given below to demonstrate the development trends in the 4th generation x86 from Intel (including 5th generation Xeon Scalable processors) and AMD, which are considered in this review.

Given the large amount of performance data available for a variety of workloads (including those mentioned above) outside of HPC and AI, the performance improvements for the Zen 4-based EPYC and 4th and 5th Gen Xeon Scalable processors discussed in this review relative to their previous generations are illustrated in a much more limited manner below.

The areas briefly considered in the review also include the problems of possible security violations that have been actively analyzed in recent times. Attention to security issues has increased due to the growth in the number of cores in processors, accompanied by the use of virtualization tools within a single processor.

Security problems, as is known, can occur for a wide variety of integrated circuits [32]. The von Neumann architecture is vulnerable to possible security violations (see, for example, [33]). In superscalar processors that support SMT, OoO and branch prediction, the possibilities of security violations increase and are realized on processors of a wide variety of architectures — x86, ARM, RISC-V [34, 35]. Although, naturally, first of all, such information appears for x86 due to its much greater prevalence (see, for example, [36, 37]).

The 4th generation of Intel and AMD server processors have special hardware features to liquidation certain security breaches. Recently, many publications have appeared on processor security issues, and detailed studies have been conducted, including of these new features (e.g., for

Intel - [38]). However, these features primarily relate to possible security breaches that occur when using virtualization features. Perhaps, these hardware improvements cannot suppress all possible security breaches in either Intel or AMD processors, since data on new types of attacks are emerging.

For different generations of AMD Zen and Intel Xeon architectures, including the 4th generation considered in the review, this is demonstrated in [39–44]. Among these publications, a greater number are devoted to attacks on different generations of AMD Zen. The corresponding improvements in processors primarily help to complicate possible security breaches and reduce the likelihood of their implementation. Hardware security enhancements, primarily for virtualization and cloud computing, are discussed further in the sections on the microarchitectures of specific processor families.

Modern x86 processors from Intel are accompanied by a number of new interesting hardware improvements, implemented in the form of accelerators. They are specialized and useful for some important, but narrowly focused areas of their application, and therefore are considered in the review rather briefly (see below for more on them in Section 3).

Next, Section 1 analyzes important general hardware and software capabilities, including those formed before the advent of the conditional 4th generation processors. Section 2 considers the architecture of Zen 4 processors and computing systems based on them, as well as their performance. Section 3 analyzes information about the architecture of various Xeon processors, classified here as the conditional 4th generation, and computing systems based on them. Since Section 3 combines data for different Xeon processors, a discussion of the performance of systems based on them, including comparison with Zen 4, is held in a separate Section 4. A short Section 5 considers information about the use of computing systems based on 4th generation x86 for virtualization and cloud technologies. Section 6 analyzes data on the latest high-performance Xeon 6 and Zen 5 processors, including initial performance data. Section 7 is devoted to building homogeneous and heterogeneous nodes of clusters or supercomputers using x86.

The appendix provides a short list of abbreviations commonly used in the review, which, in our opinion, are not very widely used in the literature.

1. General Features of the x86 Processors in Question, Including those Made Before the Advent of their 4th Generation

1.1. About Previous Generations of x86 Hardware

Due to the rapid progress of the technology used to produce processors, the need for a separate south bridge on the motherboard has practically disappeared. Since the active orientation towards SoC continues, from the author's point of view, the use of the chipset will probably disappear in modern server processors from AMD and Intel in the near future; it is no longer used for AMD processors. Intel Xeon Scalable processors of the 3rd and 4th/5th generation still have a chipset (C620A and C741 respectively).

More important functions for processors are no longer based on the chipset. If the C620A chipset also had hardware accelerators, then the C741 no longer contains them (they are integrated into the Xeon). The C741 supports work with SATA, USB, some security functions, etc. [45]. And modern EPYC processors have an integrated block responsible for all input and output data needs (meaning not just input/output, but any data transfer to and from the processor, including work with RAM). For more details, see Section 2.2 below.

Many modern server processors from different developers use a multi-chip approach with chiplets — with two-dimensional (2D) technology. When monolithic processors are placed on a silicon wafer, a single defect can render the entire processor inoperable. However, with a multi-chip approach, the defect is limited to a single die. The defective die is simply discarded, and only defect-free chiplets are assembled into a common package.

AMD was one of the first major computer companies in the world to use this technology [46], and uses this approach in its GPUs as well [2]. By reducing waste, AMD is able to offer lower prices for its products.

Intel has also been using multi-chip architectures for a very long time. And in its scalable Xeon processors, Intel has been using multi-chip technologies since the second generation [47]. However, due to certain technological limitations and the smaller number of cores used in server processors, Intel has so far used this approach to a more limited extent.

TABLE 1. Brief comparative information about the 3rd generation x86 from Intel and AMD

	Number of cores per CPU	Frequency, GHz	CPU price[1]	Number of CPUs used	SPEC CPU2017 fp_speed (base/peak)[2]	SPEC CPU2017 fp_rate (base/peak)[2]
EPYC 7763	64	2.4–3.75	$7890	2	263/270[4]	663/7101[3]
				1	177/181[5]	333/3571[3]
EPYC 7663	56	2–3.5	$6366	2	246/253[6]	596/637[4]
				1	162/162[6]	299/3201[3]
EPYC 7643	48	2.3–3.6	$4995	2	236/245[6]	576/617[6]
				1	153/158[7]	288/3041[4]
EPYC 7543	32	2.8–3.7	$3761	2	232/242[4]	531/540[4]
				1	153/158[7]	260/268[6]
EPYC 75F3	32	2.95–4	$4860	2	238/248[4]	546/565[4]
				1	160/165[5]	273/2831[3]
Xeon 8380HL	28	2.9–4.3	$13012	4	255/255[8]	667/7061[5]
Xeon 8380	40	2.3–3.4	$8978[3]	2	277/277[9]	605/640[10]
Xeon 8368	38	2.4–3.4	$7214	2	269/260[10]	586/620[10]
Xeon 8362	32	2.8–3.6	$6236	2	270/270[11]	561/589[11]
Xeon 8358	32	2.6–3.4	$4607	2	251/251[12]	507/534[10]

[1] Data as of 12.05.2024 for AMD — from [49], for Intel — from [50].
[2] Data from [51] with the maximum of "base" value presented there.
[3] Last Price at 2022-04-03 (previously the price was higher) from [52]. Data as of 12.05.2024.
Result senders and servers where these results were received:
[4] ASUS RS720A-E11 (KMPP-D32);
[5] ASUS RS520A-E11(Z12PP-D32);
[6] Cisco UCS C255 M6;
[7] Lenovo ThinkSystem SR655;
[8] Lenovo ThinkSystem SR860;
[9] xFusion 2288H V6 and ASUS RS720-E10-RS12(Z12PP-D32);
[10] ASUS RS720-E10-RS12(Z12PP-D32);
[11] ASUS RS720-E10(Z12PP-D32);
[12] Dell PowerEdge R650;
[13] ASUS RS520A-E11(KMPA-U16);
[14] Dell PowerEdge R6515;
[15] New H3C UniServer R6900 G5.
Note: To clarify the data on the servers used, the motherboard used in the benchmark is sometimes indicated in parentheses.

What was typical in the x86 performance area (primarily for HPC and AI) before the advent of the 4th generation of AMD Zen and Intel Xeon Scalable, we will describe only "in general", at the macro level, since the analysis of performance data for the corresponding x86 generations is not related to the topic of the review.

AMD has long distinguished itself from Intel in the processor market by the lower cost of its hardware, similar to that produced by Intel [48]. Some comparative data for Zen 3 and 3rd generation Xeon Scalable processors are presented in Table 1.

A very important common feature of the Intel and AMD processors considered in the review, which is given special attention in the review, is the desire of both companies to increase the bandwidth of the memory

used. The performance of microprocessors grows faster than the memory bandwidth [53], which can limit the performance of applications. This has been known about processors since the last century [54]. And since x86-processors from AMD and Intel are still used in servers more often than others, the influence of memory bandwidth is most widely manifested here.

Xeon processors and cores have historically typically outperformed AMD's corresponding hardware. AMD's previous successes with Opteron server processors were largely due to high memory bandwidth figures [55]. AMD's current successes with different Zen generations have also been coupled with higher single-core memory bandwidth compared to Intel cores [56]. Although these data only cover memory reads, they provide a reasonable estimate of per-core bandwidth [56]. Memory bandwidth for the processors analyzed in this review will be discussed later.

For widely used benchmarks and applications that do not actively use matrix multiplication and AVX-512, especially for memory-bound ones (e.g., CFD tasks), AMD's performance advantages have emerged in the second and third generations of EPYC Zen. In addition to the larger number of cores, these are related, in particular, to the aforementioned lower memory bandwidth per core in Xeon of the corresponding generations compared to AMD [56].

Messages about the preference of AMD EPYC server processors over Intel Xeon have been appearing on the Internet for a long time. References to the corresponding data on the performance advantages of AMD processors in widely used benchmarks and applications are given above in the Introduction. The data in Table 1 also provide an additional general illustration. For an example of comparative cost and performance estimates, we also point to [52].

Note that when analyzing performance, it is also important to take into account the dates of obtaining the corresponding results (which is related to the versions of the SDKs that became available and were used). Thus, in Table 1, the Xeon Platinum 8380 slightly outperformed the EPYC 7763 in the SPECcpu2017 fp_speed benchmark. But this data in the table for the Xeon 8380 (as well as in the SPEC CPU2017 fp_rate benchmark) was obtained in 2023, and for the EPYC 7763 — in 2021. And the maximum result of the Xeon 8380 in 2021 was 241/244 for the base and peak variants of the fp_speed benchmark, that is, lower than that of the EPYC 7763.

As another example, performance data from single cores to servers (including Xeon ICL and AMD EPYC Zen 3, Milan) and small clusters in a set of computational chemistry tasks using well-known applications typically show an advantage of some EPYC Milan models over older Xeon ICL models [57].

TABLE 2. Performance comparison of 2P servers based on Xeon ICL and EPYC Zen 3

Performance	Xeon 8380 (40 cores)	EPYC 7763 (64 cores)
DGEMM (GFLOPS)	**3824**	3046
DGEMM per core (GFLOPS)	**47.8**	23.8
HPL (GFLOPS)	1713	**2176**
HPL per core (GFLOPS)	**21.4**	17.0
FFT (GFLOPS)	**76.4**	54.7
FFT per core (GFLOPS)	**0.96**	0.43
GROMACS (ns/day)	133.3	**169.9**
NWChem (seconds)	**19.2**	26.7
OpenFOAM (minutes)	6.8	**6.6**

Data from [8].
Higher performance results are marked in bold.

Very useful for HPC data [8] are distinguished by their wide coverage of different areas. Some of this data is presented in 2.

These data are important because they clearly demonstrate what has developed in the 3rd generation of x86 processors from AMD and Intel: the advantage of the latter is due to the use of AVX-512 in DGEMM, which is already is eliminated in HPL (High Performance Linpack) due to the larger number of cores in AMD Zen 3. In addition to the data on three benchmarks conducted from HPCC [58] (DGEMM, HPL, FFT–FFTW3[1]), 2 shows the data of benchmarks known for highly efficient parallelization of computational chemistry applications — molecular dynamics (GROMACS with a system of about 82 thousand atoms) and quantum chemistry (NWChem, with a small system — Au+ ion, by the HF method followed by MP2 and Coupled Clusters).

Another benchmark given is the OpenFOAM (Open Source Field Operation And Manipulation CFD ToolBox), which is related to the field of continuum mechanics (motorBike — calculation of incompressible air flow around a motorcycle). Although OpenFOAM is a set of tools in C++ for solving systems of partial differential equations in space and time in discrete form, and can be used not only in the field of continuum mechanics. But it was initially aimed at CFD, and therefore further in the review it is considered together with CFD applications.

[1] https://www.fftw.org/, accessed 23.11.2024.

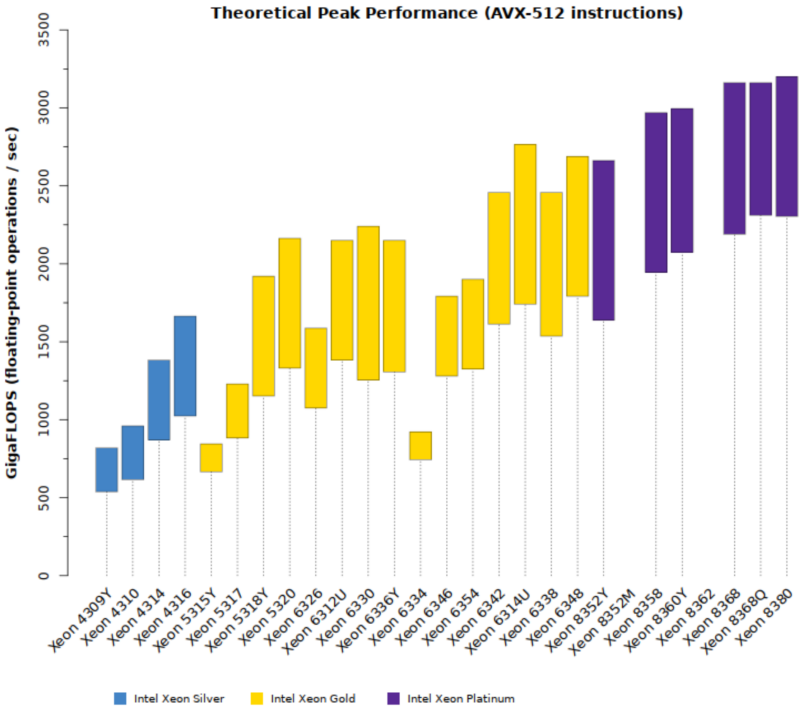


Figure 1. Peak performance of different Xeon ICL models using AVX-512 (figure from [59])

Comparison with ARM processors in [8] showed the advantages of x86 in performance; ARM may be ahead in energy efficiency. And relative performance depends heavily on applications: GROMACS and OpenFOAM were faster in EPYC, and NWChem in Xeon ICL. Accordingly, one should carefully look at the data of which benchmarks are being discussed.

But overall, it can be said that in the data presented here, EPYC Zen 3 clearly more often turned out to be more productive than Xeon ICL.

As an example of general weighted recommendations for HPC and AI on the selection of 3rd generation Xeon and EPYC processors taking into account the price/performance ratio, we refer to the data from the Microway website - [59] and [60]. For illustration, we also provide peak performance for different Xeon ICL models — Figure 1, Figure 2 and for Zen 3 (Milan) — Figure 3.

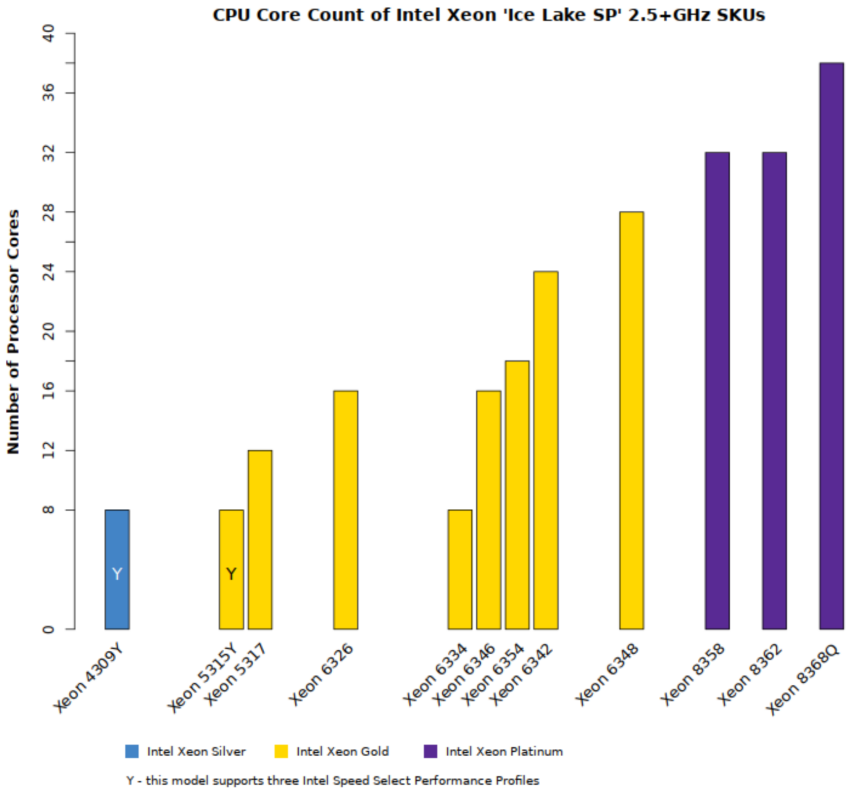


FIGURE 2. Peak performance of different Xeon ICL models using only AVX2 (figure from [59])

In Figure 1, the peak FP64 performance data for different Xeon ICL models is for AVX-512, while in Figure 2, it is for AVX2-only operation. In Figure 3, for Zen 3, the peak performance of different EPYC models (which only support AVX2) is calculated for base frequency, with theoretical data for turbo frequency shown as the tops of the dotted lines.

Despite the above-mentioned performance advantages of EPYC processors of previous generations of the Zen architecture, in reality, judging by the activity of x86 use on supercomputers in the June 2024 version of the TOP500 [1], there is no preference for using x86 from AMD in the first 50 positions of this list: EPYC was used in 22 systems, Xeon — in 27. Statistics for the entire list [61] indicate the use of the 2nd generation of Xeon Scalable processors (Cascade Lake) in 123 supercomputers from

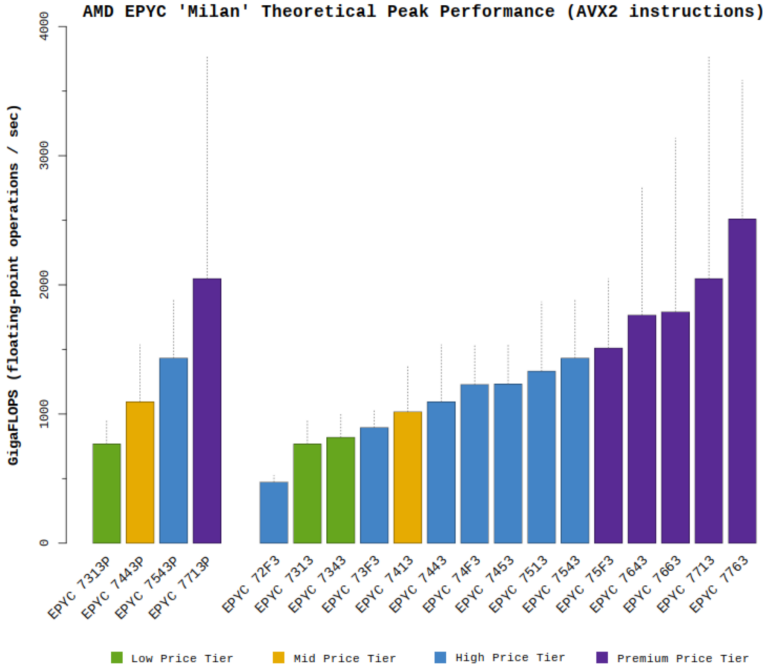


Figure 3. Peak performance of different EPYC Milan models (figure from [60])

the TOP500, and the 2nd generation EPYC Zen 2 — in 68 supercomputers. The 3rd generation Xeon ICL processors were used in 42 supercomputers on this list, and EPYC Zen 3 Milan — in 73 supercomputers. Finally, 4th generation Xeon SPR processors were used in 31 supercomputers, and EPYC 9004 in 15.

It is also interesting to note that out of 123 supercomputers with Xeon Cascade Lake, 79 used Xeon Gold 62xx, and the older Xeon Platinum 82xx and 92xx series were used on the almost twice as few number of supercomputers. The problems of selecting x86 models for building clusters and supercomputers are discussed further in a separate Section 7.

The performance review is further focused mainly on the 4th and 5th generations of Xeon Scalable processors and the 4th generation of Zen (EPYC) 9004 series — as modern, became available at about the same time, which have many data on the achieved performance. Therefore, they are combined in the review into one common conditional 4th generation of x86

server processors. Intel and AMD, naturally, provide comparative data on the growth of the performance of these processors relative to their previous generations.

Accordingly, the performance comparisons will also include data for the 3rd generation of server processors (AMD Milan and Intel Xeon ICL). And Section 6 also analyzes the initial data on the next generation of x86 from Intel and AMD — Xeon 6 (Granite Rapids) and Zen 5, including their performance. But we consider the performance comparisons of the conditional fourth generation to be the basic ones, which is why the title of the article says "around the conditional 4th generation".

1.2. About CPU Frequency Adjustment and Using it in Linux

Energy consumption issues have become increasingly important in recent years, and for computing systems this can be expressed in two aspects: optimizing the cost/performance ratio or considering energy consumption as a constraint. This is analyzed in software and hardware, including modern x86 server processors, in [62]. Here, only one important issue directly related to energy consumption is considered: regulating processor frequencies.

The discussion of this, including frequency adjustment in Linux, is moved to a separate section for four reasons.

1. Naturally, because of the impact on power consumption.
2. frequency adjustments can occur for all processors, for example, ARM, and not just for x86 server processors from AMD and Intel.
3. Because the information about the reduction of increased (turbo) clock frequencies under intensive computational load on processor cores (especially when working with AVX-512) is important, since this also affects possible estimates of peak performance.

   The information provided by Fujitsu about its servers with Xeon SPR processors indicates that using the turbo frequency mode can also lead to a decrease in the achieved performance compared to disabling this mode on computationally intensive applications, for example, with workloads at the level of the HPL benchmark [63].

   In addition, information from the manufacturer about reducing the clock frequencies of processors was often unavailable or confidential. The first generation of Xeon Scalable processors used the concept of "AVX-512 core base frequency" (for this and for reducing the frequency in turbo mode, see [64]). In the latest generations of processors considered in the review, the clock frequency supported by all operating cores simultaneously has often been indicated.

The 3rd generation Xeon introduced the concept of high-priority and low-priority cores, for which different frequencies are specified. This makes it possible to configure the cores with higher turbo frequencies provided to priority cores compared to low-priority ones, and is related to Intel Speed Select technology [65]. Although this technology itself is also supported in Xeon 6, there are no indications of high- and low-priority cores in their specifications [66]. This is not discussed in more detail here, since it only applies to certain Xeon processors, and for HPC tasks using parallelization, where changing frequencies when working with AVX-512 is most important, it seems less relevant.

4. In Linux the adjustment of processor frequency was made quite a long time ago (compared to the times of the appearance of new generations of x86), with an orientation towards a universal approach, not aimed at specific families of processors. Another thing is that the implementation of this using also specific Linux kernel drivers from the developers of the corresponding processors gave certain limitations (and many processors did not allow this to be used at all). But recently, due to the growing importance of energy consumption, the situation has changed for the better.

This section of the review provides general approaches to adjusting the frequency of different processors and terms used when working in Linux. This is used further in the sections on specific x86 processors from AMD and Intel. And the fact that the orientation in this section is made on Linux is understandable, since this OS is the main one when working with servers, in particular, with computationally complex HPC and AI applications.

The Linux kernel has CPUfreq facilities for increasing or decreasing the processor frequency to improve performance or save power, and its governors for the above task [67, 68].

In [68], there are 6 different "target direction" governors — levels of abstraction of the processor operation, in which the corresponding algorithms are used, and between which one can choose when working in Linux.

To analyze this, the C- and P-states of the processor are used, which have several levels each. They have long been used in studies of multi-core processors (see, for example, [69]). These processor states, like also the T-states, are related to the ACPI (Advanced Configuration and Power Interface) interface [70].

T-states refer to the mechanism for reducing the clock frequency when the temperature reaches a critically high value. This seems natural for Intel processors with the implementation of AVX-512. But Intel with its Xeon Scalable processors has long offered its own efficient mechanisms for managing P-states (see, for example, [71]). AMD does not mention T-states in the description of the architecture of Zen 4 processors [23], in which AVX-512 is implemented in a simpler two-stage mode (see Section 3.1 below), and we do not consider them in this section.

There are some refinements/changes in specific processor families, which are discussed in other sections below. Current dynamically changing information, including the corresponding Linux drivers for different processors, can be obtained from [72]. The above processor states are used in the corresponding Linux distribution descriptions, and we base here on the information for widely used HPC distributions [73, 74].

C-states. The greater the value of n in the Cn state, the "further" the processor is from the operating state and the less energy it consumes.

C0: the processor is on and running.

C1: is the first idle state. The processor does not execute instructions. Stops CPU main internal clocks via software [73], but is typically not in a low-power state, and can continue to operate with little or no delay [74]. In [74], this state is briefly called Halt.

C2: the main internal clocks of the processor is stopped via hardware. But the software state remains normal, and operation can be resumed after interrupts, but with a certain delay. In [74] this state is briefly called Stop-Clock.

C3: The processor is asleep and all its internal clocks are stopped. It does not have to maintain cache coherence [73]. It takes significantly longer to wake up than from state C2. In [74], this state is briefly called Sleep.

States C0 and C1 are supported by all processors [73], states C2 and C3 are optional.

P-states. When a processor is running (in state C0), it can be in different performance states (P-states). All P-states are operational, and are related to the frequency and voltage of the processor. The larger the value of n in state Pn, the lower the frequency and voltage at which the processor operates. Although P0 always refers to the

highest performance state (there are some special features for turbo mode), the number and implementation of different P-states depend on specific processors [73].

Accordingly, information about the C- and P-states of the processors analyzed in the review will be discussed further in the corresponding sections of the review.

To conclude this section, let's look at the CPUfreq governors themselves in Linux.

performance: Forces the processor to set a static maximum possible frequency for it. This is intended for cases where the processor is heavily loaded almost all the time, and there should be no energy saving.

powersave: The processor frequency is being installed to the lowest possible value. In this case, energy is maximally conserved, and performance is reduced. However, as noted in [73, 74], this does not always correspond to the desire to reduce the energy used as much as possible. The corresponding frequency values themselves can be changed in Linux [73].

onedemand: This governor is aimed at dynamically changing the frequency depending on the current system load [68]. It is clear that changing the frequency always leads to certain delays, and with sufficiently frequent switching between standby and high workload modes, the use of this governor is unacceptable. But it is suitable for situations when the system is busy only at certain times of the day. As indicated in [73, 74], at high system load the governor will set the maximum processor frequency, and at low load — the minimum.

conservative: This governor is similar to ondemand and is also designed to dynamically change frequencies. However, it does not do so by jumping from minimum to maximum frequency, but more smoothly [68]. This governor adjusts to the clock frequency that it considers most suitable for the load, rather than simply choosing between maximum and minimum. Although this can provide significant power savings, it requires a larger delay than the onedemand governor [74].

userspace: This governor allows a process or program to set configurable values in sysfs file for the frequencies used [68] and can provide a more efficient balance of performance and power consumption [74].

There is also a 6th governor, *schedutil* [68], but it is not considered at all in [73] and [74], and will not be analyzed here. The operation of governors presupposes not only a certain version of the Linux kernel, but also the corresponding drivers implementing this operation for specific processor families, which leads to the appearance of their own peculiarities. Therefore, as already mentioned above, further in the review, frequency regulation is considered for specific generations of the analyzed AMD and Intel processors.

1.3. Software Development Kit (SDK) for x86

1.3.1. *Traditional (Classic) and New SDKs for x86 from Different Developers*

Due to the vast prevalence of x86 usage, many different SDKs have been created and used over the years that apply to varying degrees to a wide variety of x86 processors from different manufacturers, and are therefore discussed here as part of Section 1 on general x86 processors.

SDKs for x86 are the most widely used and the most well-known in the world, which are regularly analyzed in publications, and their effectiveness not only depends quite strongly on the orientation, for example, of a specific application being created using them, but also changes rapidly over time with the appearance of new versions of the SDK. Their analysis should include not only the achieved level of optimization, but also the breadth of coverage of available capabilities (for example, in compilers — support for new versions of OpenMP or language constructs of new language standards).

Therefore, only a brief analysis with basic and practically useful information about compilers relevant for HPC, applicable for the processors considered in the review, and very little about the main program libraries is carried out here. Debuggers and profilers are not considered in the review at all. And libraries of parallelization tools and, accordingly, communication by messages of the MPI type are almost not mentioned at all.

In the following, we are talking about compilers for C/C++ only (generally speaking, also Fortran, but in this case the differences are mainly in the front-end part of the compilers). Due to the ultra-fast changes in the SDK, almost all references cited here are limited to publications of recent years.

Table 3. Current versions of x86 compilers relevant for HPC

Compiler	GNU[1]	LLVM[2]	Nvida[3]	Cray[4]	AMD[5]	Intel
C/C++	gcc/g++	Clang	nvc/nvc++	craycc	aocc	icc
Fortran	gfortran	Flang	nvfortran	craftn	aocc	ifort

Current versions as of 09/03/2024: [1] gcc 14; [2] LLVM 19.1.0-rc1; [3] NVHPC 24.7; [4] CPE 24.07; [5] aocc 4.2.

SDKs from different manufacturers can commonly be used for both Xeon and EPYC (some limitations of this will be discussed below). Intel's SDK has long been considered one of the best for x86 in the world, and in practical terms it is important to note the availability of these tools for use with EPYC. But recently Intel offered a new original generation of SDK, oneAPI [75]. Currently, both the classic Intel SDK ("classic" is an Intel term) and the oneAPI tools are used in the world. oneAPI software tools are a huge new product from Intel that requires a separate analysis, and therefore will be considered in the next subsection. Here we will limit ourselves to traditional, relevant for HPC and well-known SDKs from other developers with mentions of new SDK versions and developments. Such compilers, which can be used from PCs to supercomputers, are presented in Table 3.

The gcc 14.2 compiler supports[2] OpenMP 4.5, partially supports OpenMP 5.0, some OpenMP 5.1 capabilities, and some capabilities from OpenMP 5.2. The gfortran compiler supports[3] the Fortran 2008 standard (and, accordingly, the PGAS coarrays model) and some capabilities of Fortran 2018 and also gfortran supports OpenMP 4.5, some capabilities of OpenMP 5.1. Preparations for support of the new Fortran 2023[4] standard are also beginning.

References to various components of the AMD SDK for working with processors of various Zen architectures are public available[5], and a general guide to the SDK for HPC tasks with high-performance EPYC 9004

[2] https://gcc.gnu.org/wiki/openmp, accessed 3.09.2024.

[3] https://gcc.gnu.org/onlinedocs/gfortran/Standards.html, accessed Accessed 3.09.2024.

[4] https://gcc.gnu.org/gcc-14/changes.html, accessed 3.09.2024.

[5] https://www.amd.com/en/developer/zen-software-studio.html, accessed 30.10.2024.

processors is presented in [76]. There is a similar SDK document for the less high-performance EPYC 8004 processor series, which also uses the Zen 4 architecture (they are almost not considered in the review). AMD notes in its aocc compiler set that they are primarily focused on processors of different generations of Zen and HPC tasks [77]. The aocc 4.2 compilers are based on LLVM 16.0.3. It supports C/C++ 17 and Fortran 2008 (but without coarrays tools). OpenMP 5.0 is supported for C/C++, OpenMP 4.5 for Fortran; For more details see [78].

For AMD, it is worth mentioning the new tools under development — a compiler based on LLVM[6] and AOMP[7] — an open source compiler based on Clang/LLVM, focused primarily on the use of OpenMP tools on AMD GPUs. For the latest EPYC Zen 5 Turin processors, aocc 5.0 was developed based on LLVM 17.0.6 [79]

In relation to HPE/Cray, the data below is not about the traditional versions of their compilers, which are no longer supplied, but about new ones. Currently, HPE offers SDK tools called CPE (Cray Programming Environment). One of the CPE components, called CCE (Cray Compiling Environment) [80], includes, in particular, compilers that support Fortran, C/C++ and UPC (Unified Parallel C) languages. The Fortran compiler of HPE Cray almost completely supports the Fortran 2018 standard with some exceptions. Modern versions of HPE Clang C/C++ are based on Clang/LLVM. And the common CPE software tools also support several third-party compilers: aocc, Intel, GNU and NVIDIA compilers [81].

Mathematical libraries, MPI/SHMEM messaging facilities and other CPE components are not discussed here (see [82] for more information).

NVHPC 24.7 [83] contains the nvc compiler with support for ISO/ANSI C11, nvc++ compiler with support for ISO/ANSI C++17, and nvfortran compiler with support for ISO/ANSI Fortran 2003. Extensions to these compilers to support CUDA are not discussed here.

Of the Intel compilers, we will only point out the classic icc and ifort compilers, which are actively used worldwide to translate programs executed on a variety of x86 processors, not only Intel but also AMD. It is often assumed that the highest level of optimization can be achieved when using these Intel compilers, although there is evidence that this is often not the case, especially when using these compilers for calculations on x86 from AMD.

[6] `https://github.com/ROCm/llvm-project`, accessed 2.05.2025.

[7] `https://github.com/ROCm/aomp`, accessed 2.05.2025.

Studies comparing the optimization level achieved by these compilers for x86 processors began a long time ago, and are still actively carried out today. Let us give several examples. Various compilers and the relevance of their use for AMD Zen 2 (Roma) were considered, for example, in [84]. Also, for calculations on EPYC Roma and Milan, optimization by five different compilers, including icc and aocc, was studied in [85]. Here, the Intel compiler showed the best results, ahead of aocc, although the difference in the achieved performance was usually not that great.

As another example, we point out a recent paper [86], in which icc, icx (this is a new generation, from Intel oneAPI), aocc, gcc and clang were investigated in MD-bench benchmarks for molecular dynamics. Different versions of the same compiler are also compared (for example, gcc in [87], but there the comparison was carried out on an Intel Core i7 processor).

All of these papers included optimization data for Intel and AMD processors of the 3rd generation or earlier. And the rapid advances in the hardware of the processors reviewed in this review and the ISA extensions in them, such as AVX-512 support in EPYC Zen 4, make the data actual for the specific x86 generations discussed in this review.

For the EPYC 9004 (EPYC 9654 model), the performance data for the HPL benchmark of the aocc compilers, aocl (AMD Optimizing CPU Libraries) and OpenMPI compared to oneAPI Base & HPC Toolkit Classic Compiler 2022.2.0, oneAPI MKL 2022.2.0 (see below in Section 1.3.2) and Intel MPI 2021.6 are presented in [88]. In [89], using the EPYC 9654 as an example, the performance obtained from the aocc 4.0 and Intel compilers was compared for a number of well-known applications, including molecular dynamics, CFD and weather prediction. The achieved difference was found to be within a few percent.

In [90], the achieved performance of the well-known CFD application ANSYS LS-DYNA on a dual-processor server with EPYC 9654 was compared using the aocc 4.0.0 and ifort 2019 compilers. When compiling with AVX2 support, aocc produced slightly faster executable code, and when using AVX-512, the ifort compiler produced slightly faster code.

A wide range of different applications were used to compare optimizations between gcc 13.1 and LLVM Clang 16.0 compilers [91]. Although the difference in the geometric mean estimation across all applications was only a few percent, the well-known mini-application of molecular docking

miniBUDE worked 1.6 times faster when using gcc, and the oneDNN benchmark (the Intel oneDNN library is discussed below) worked several times faster when compiled with Clang. And the molecular dynamics application LAMMPS had similar performance when using both compilers.

The use of the geometric mean estimate is partly limited by the uncertainty of the level that the various "contributions" should have. In any case, possible refinements of the optimization parameters used for specific applications may be important. Therefore, at least for freely available compilers, it is advisable to provide the programmer with the possibility of choosing them.

Due to the ISA extensions in Intel processors, starting with Xeon SPR, some modern software tools that use these extensions will not be able to run on AMD Zen 4. And some SDKs from oneAPI are also not applicable on Zen 4, for example, the oneDNN library (see Section 1.3.2 for more information) for working with AI tasks, which supports the AMX ISA extension (see Section 3.1.2 for more information).

However, AMD offers its own similar library ZenDNN, optimized for the characteristic features of Zen 4 — a large number of cores and a large L3 cache capacity [**92**]. Data on the performance achieved with ZenDNN on Zen 4 is available, for example, in [**93**].

1.3.2. *OneAPI Tools and Intel OneAPI Software Toolkit*

Due to the very wide scope of the application area of oneAPI tools, they are discussed here in a separate subsection. Of course, oneAPI tools deserve a much more detailed consideration than will be given below. However, a little has been said above about their "classic" part from Intel, and a large and very important part of the oneAPI components relate to AI tasks, which are less addressed in the review compared to HPC.

From the author's point of view, the most striking achievements of Intel in recent times lie precisely in the area of software, SDK. Here it is necessary to distinguish between the oneAPI project (an open source project with an open and standards-based set of interfaces, oriented towards different types of architecture, including not only processors, but also accelerators), implemented on the original initiative of Intel, supported within the Unified Acceleration (UXL) Foundation group [**75**], and the specific optimized free available software product Intel oneAPI for different hardware.

The UXL oneAPI project is largely based on the SYCL standard, a C++ extension that is currently most relevant for use in various accelerators (e.g., GPUs), although it can also be used on modern processors. Intel proposed and implemented an extension to SYCL, DPC++. Some information on the performance achieved using these tools in GPUs from different developers can be found in [2]. In addition to DPC++/SYCL, oneAPI includes a wide range of libraries with unified standardized interfaces, which is where the oneAPI in the name comes from. Naturally, only the x86-specific Intel oneAPI tools are considered here. Everywhere below, oneAPI refers specifically to Intel oneAPI for x86.

Of course, oneAPI includes analogs of famous classic Intel software tools. For example, the MKL library is now called oneMKL. It is known to include, in addition to programs for vector mathematics, dense and sparse linear algebra, also tools for the Fourier transform and random number generators. Classic C++ and Fortran compilers are also part of oneAPI, but further development will be carried out on oneAPI compilers based on LLVM. This does not mean that all new oneAPI compilers always surpass classic Intel compilers in the achieved level of optimization (see [94]), but this may depend on the version.

Intel is actively focusing its 4th and 5th generation Xeon Scalable processors on solving AI problems, and accordingly, oneAPI provides libraries relevant to this area. For example, we will mention the oneAPI Data Analytics Library (oneDAL) for machine learning, which accelerates big data analysis at all stages: preprocessing, transformation, analysis, modeling, validation, and decision making. And oneAPI Deep Neural Network Library (oneDNN) provides the basic building blocks used in deep learning applications.

We will only provide references to documentation on oneAPI tools for HPC [95] and for AI [96]. Although Intel's work is increasingly focused on AI tasks, it is worth noting Intel's membership in the OpenHPC community, which is focused on bringing together common open source software components needed to deploy and manage Linux clusters [97].

The current (available as of 5.09.2024) documentation for the new Intel compilers is 2024.2.1 for DPC++/C++ and 2024.2.0 for Fortran (ifx), while for the classic compiler it is 2021.1.0 for icc and 2024.2 for ifort. Full

descriptions of oneAPI are available in [98]. As for ifx, there is data on a performance increase of up to 17% when used in a 2P server with a Xeon Max 9480 compared to the classic ifort (in Fortran-based benchmarks from SpecOMP 2012) [99], although in some benchmarks there was a 3% decrease.

When talking about using the Intel SDK for working with EPYC Zen 4, it should be noted that the highest performance indicators of AMD Genoa processors (Zen 4, containing the highest-performance cores — see below in Section 2.2) in the SPECcpu 2017 benchmatrks [51] were usually obtained using aocc tools. However, in supercomputer centers, including those containing EPYC Genoa in nodes, Intel compilers are probably used more often. But the optimal and often used in supercomputer centers is to offer a whole set of different compilers.

As for math libraries, in addition to using open-source libraries, MKL can also be used with Zen 4. The MKL library for a 2P server with EPYC 9334 in [100] for HPL gave 5% higher performance than the open-source OpenBLAS library. And the well-known disabling of the processor ID check in MKL on this server gave a 20% performance increase.

In [88], on a 2P server with EPYC 9654, higher HPL performance was obtained using aocc, aocl, and OpenMPI than using oneAPI Base & HPC Toolkit Classic Compiler 2022.2.0, oneAPI MKL 2022.2.0, and Intel MPI 2021.6. Although library performance may have been the determining factor here, this result may be also due to the lack of ID checking disablement for Zen 4.

2. 4th Generation AMD EPYC Processors (Zen 4 Architecture)

Zen 4 naturally has a lot in common with the architectures of previous generations of AMD processors, especially with Zen 3. It is clear that many of the striking features of Zen 4 are due to AMD's long-standing use of multi-chip technology. But here we will point out only one common feature that has long been available in x86 server processors from both AMD and Intel — support for SMT (specifically, two logical cores on one physical core). For improvements due to the use of SMT in Zen 4 compared to Zen 3, see [22]. Further, SMT will be discussed mainly only in Section 2.4 when considering the impact of SMT on performance. For many HPC tasks, enabling SMT is ineffective.

2.1. Microarchitecture and ISA of Zen 4 Cores

We will begin our discussion of processor microarchitecture with the core microarchitecture. This analysis is important in a broader sense than the topic of this review, since these cores are also used in other (non-server) AMD processors. And achieving high performance per core is important not only because of the desire for high performance of the entire processor, but especially for situations where the software license is paid for by the number of cores used. The information in this section will also include data on improvements in ISA, since instructions are executed by the cores of the processors.

The main AMD publication on the Zen 4 core microarchitecture can be considered the work [**101**]. Each core includes 1 MB of private L2 cache, which is twice as much as in Zen 3. The IPC improvement compared to the previous generation for single-threaded desktop applications is on average 13%. The Zen 4 core can operate at a frequency of 5.7 GHz. Relative to Zen 3, both single-threaded performance (by more than 29%) and performance per watt in multi-threaded workloads are improved [**101**].

The construction of the front-end part of Zen 4 (see the block diagram in Figure 4) is essentially not fundamentally different from the block diagram for Zen 2 on the well-known site wikichip [**102**], where detailed information on the microarchitecture of processors is posted. But in [**102**] for Zen 2 the figure is more detailed and covers a larger part of the core. Similar information for Zen 3 and Zen 4 in [**103**, **104**] is considered in less detail.

Zen 4 can dispatch up to six integer operations per clock, and execute up to three loads and up to two stores. Branch prediction accuracy has been improved compared to Zen 3. Buffer sizes throughout the core have also been increased. The capacity of the pre-decoded instruction cache (Op Cache in Figure 4), the capacity of retire queue, and the integer and floating-point register files have all been increased [**101**].

As for the front-end part, in [**22**] concerning the instruction fetch engine, an increase in the throughput of the instruction fetch queue and their dispatching, as well as the capacity of the micro-operation cache, is noted. Some of this data is displayed in Table 4. And on the slide related

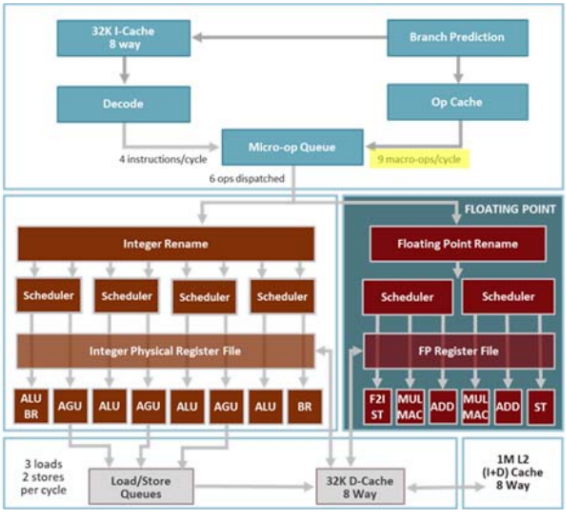


FIGURE 4. Microarchitecture of Zen 4 cores (figure from [101])

to the execution units in [22], an size increase of the load/store queues, L1 and L2 DTLB, etc. is noted (see also Table 4).

We will talk about execution units below, since Zen 4 expanded the ISA and introduced partial support for AVX-512. Interest in more detailed data on the Zen 4 microarchitecture than has become freely available, including detailed block diagrams, has led to the emergence of works on establishing the relevant information using microtests (see [106]). Detailed quantitative data on the Zen 4 microarchitecture is available on the wikichip website [104]; they were partially used in constructing Table 4.

Naturally, the increased performance of Zen 4 cores relative to Zen 3 cores is also affected by the increased capacity of the L2 cache (see Table 4) and the L3 per core cache (L3 is common for all cores, this will be discussed below).

In the overall plan, we note that the above-mentioned and Table 4 improvements to the Zen 4 microarchitecture relative to Zen 3 provide an IPC increase of 14% [22, 23, 107]. The data on which improvements to the core microarchitecture contribute what amount to the IPC growth was

Table 4. Zen 3 vs Zen 4 Cores Comparison

	Zen 3	Zen 4
Load queue entries	72	88
Storage queue entries	64	64
Micro-op cache size	4096	6912
L1 I/D-cache 8-way set-associative, KB	32/32	32/32
L2 cache 8-way set-associative	512	1
L3 cache/core MB[1]	4	4
L1 DTLB entries[2]	64	72
L2 DTLB entries[3]	2048	3072
L2 cache latency, cycles	from 12	from 14
L3 cache latency[4], cycles	46	50
Issue width (INT+FP/SIMD)	10+6	10+6
Integer physical registers file entries	192	224
Integer scheduler	96	96
FP registers file entries	160	192
Reorder buffer entries,	256	320
FADD/FMUL/FMA[5] latency, cycles	3/3/4	3/3/4
L1 BTB entries	2×1k	2×1.5k
L2 BTB entries	2×6.5k	2×7k1

[1] 16-way set-associative;
[2] fully associative;
[3] Zen 4: 24-way set-associative vs Zen 3: 16-way set-associative
[4] load-to-use (average); BTB — branch target buffer.
[5] floating-point vector instructions Add/Multiply/Fused Multiply-Add
Data from: [103, 104]; [105].

presented on a slide in an AMD report that was initially confidential but then became freely available on a number of websites (see, for example, [108]). But these data with a geometric mean estimate for 22 different workloads apply to the Zen 4 cores in AMD Ryzen processors, and it is obvious that the corresponding typical PC workloads were also used.

According to these data, the smallest contribution to IPC growth is made by the L2 cache capacity, almost the same by the execution unit, twice as much by load/store, almost the same by branch prediction, and next two times more by other units of the front-end part of the cores.

To these IPC data, one can also add the similar estimate of IPC growth (by 12%) obtained in [88] for EPYC 9654 relative to Zen 3 Milan-X (EPYC 7773X) in the well-known quantum chemistry application Gaussian 16.

The execution units in Zen 4 have been significantly upgraded compared to Zen 3 due to the introduction of support for the ISA extension, AVX-512, which has long been used in Xeon processors and previously gave them clear advantages in peak performance. This fact was simultaneously accompanied until very recently by other facts — the lack of hardware support for 512-bit vectors in both AMD processors — in EPYC, and server ARM processors, including Nvidia Grace with the Neoverse V2 architecture (A64FX are an exception in this regard [**7**]).

There are obvious reasons for this — the need for additional chip area due to AVX-512 support, and accordingly an increase in cost and power consumption. And the developers, naturally, compared the possible increase in processor cost with the increase in productivity for widely used applications, including in the HPC area. Although for applications whose performance increases significantly, working with AVX-512 can also provide an increase in energy efficiency.

The most obvious, from the author's point of view, where it is very useful to have 512-bit vectors for the area of HPC is large dense matrix multiplication (DGEMM), which is actual, for example, for explicitly correlated quantum chemistry methods (but not for the widely used quantum chemical DFT methods). And the most famous benchmark where matrix multiplication is important usually indicate HPL [**109**].

In Zen 4, AMD implemented hardware support for a part (from the author's point of view, the most important) of the "subsets" of the full Intel AVX-512 extension (for these subsets, see, for example, [**110**]).

In addition to the native for HPC implementation of AVX-512 fused multiply-add (`FMA`) instructions in FP64 format, Zen 4 supports the low-precision BF16 format (often considered more useful than FP16), which is relevant for AI. Another key support for AVX-512 implemented in Zen 4 for AI tasks is the execution of Vector Neural Network Instructions (VNNI) [**22**, **23**]. The other AVX-512 capabilities supported in Zen 4, including those for cryptography, are not discussed here at all.

To discuss the hardware implementation of AVX-512 in Zen 4, we need to look at how it was implemented earlier in Xeon.

In terms of comparing the possible performance when working with AVX-512 in Xeon, which has long supported such capabilities, it should be borne in mind that their use with FMA operations for HPC tasks led to a strong decrease in the used clock frequency (compared to the turbo frequency specified by Intel in the specification) immediately upon executing such instructions, and not after increasing the energy allocated by the processor. The lack of a detailed description by Intel of the mechanisms for reducing the frequency when working with AVX-512 led to the emergence of studies of this (not necessarily in the form of scientific publications) and works aimed at more efficient use of AVX-512, despite the frequency reduction (see, for example, [**111**–**113**]). Naturally, the energy efficiency achieved in this case also began to be studied [**114**].

However, there is information about a radical improvement in the frequency reduction mechanism for Intel ICL desktop processors [**115**]; in Xeon ICL, the frequency reduction mechanism has also been improved (see, for example, [**116**]). But as of the end of 2023, Intel's detailed data on the behavior of turbo frequencies when working with AVX-512 were classified as confidential [**117**].

The advisability of hardware support for matrix operations by processors, especially for the HPC area, is generally questionable [**109**]. And the question arises about the greater efficiency of using GPUs instead of processors for complex calculations in applications where performance increases significantly when working with 512-bit vectors. However, AMD, with a focus on HPC (and AI) tasks, has provided support for AVX-512 in its new Zen 4 processors.

AMD's hardware implementation of the AVX-512 extension is significantly different from its implementation in Xeon Platinum and most Xeon Gold models. AMD has chosen the path of a economic solution, which may not require a strong reduction in clock frequencies when working with AVX-512, as in Xeon. However, the peak performance of the cores (due to the lower FLOPS/clock in Zen 4) is still lower than Intel's.

AMD's basic information on how AVX-512 support was implemented in hardware in Zen 4 is presented in [**22**, **23**]. Although Zen 4 has a set of 512-bit registers required for AVX-512, it then uses 256-bit channels

and 256-bit execution units. And the execution of 512-bit operations is implemented as two consecutive 256-bit operations performed in two adjacent clock cycles. From the developers' point of view, such an implementation contributes to high energy efficiency [**22**].

There are also suggestions that EPYC 9004 can spend more time in turbo mode [**100**]. And, naturally, it becomes possible to execute binary code of applications using AVX-512 on Zen 4.

But AMD has beefed up its AVX-512 execution units even more. Xeon Platinum and most Xeon Gold cores have two units capable of 512-bit `FMA` operations, which for FP64 gives 32 FLOPS/clock. This seems natural for DGEMM matrix multiplications, which are relevant for HPC, and can achieve performance that is closest to the peak value.

Each Zen 4 core, like Intel Xeon, has two execution units, but their hardware contains an additional vector addition unit to the `FMA` [**22**]. Accordingly, each physically 256-bit execution unit produces 3 results over vectors of length 4 FP64 numbers per clock cycle, and two execution units provide 24 FLOPS/clock cycle.

This does not seem natural for the implementation of DGEMM, and does not contribute to achieving performance in DGEMM close to the formal peak value. One can make assumptions about how such advanced capabilities of Zen 4 can be effectively used. According to [**100**], in a single-processor (`1P`) configuration in the HPL benchmark, this made it possible to achieve performance above the peak value calculated without taking into account the presence of an complementary addition unit (i.e., 16 FLOPS/clock).

Section 2.4.3 presents data from AMD confirming this level of achievable performance for a dual-processor (`2P`) configuration. Additional research is needed to clarify the situation. Data on the performance of EPYC 9004 in DGEMM and HPL are discussed further in Section 2.4.3.

The FLOPS/clock performance gap between Zen 4 cores and 4th and 5th generation Xeon Scalable processors can be offset by higher frequencies, higher core counts, or other Zen 4 benefits — but that's refers to real-world performance, which will be discussed later.

In Xeon SPR, Intel has already implemented a new ISA extension, AMX (Advanced Matrix Extensions) [118], with support for low-precision matrix operations for AI tasks. This is a major advantage for these Xeon processors in terms of AI tasks. As for HPC, AMX could potentially become relevant on a very limited scale — if it can be effectively used to emulate higher-precision calculations. Work in this direction is currently underway, but mainly with a focus on tensor cores in GPUs — see, for example, [2, 109].

In conclusion of this section, it should be noted that AMD also has slightly different Zen 4c processor cores. Containing up to 8 Zen 4c core complexes (CCX), the core complex has a shared L3 cache of 16 MB, which is half the size of Zen 4 cores [23]. CCX core complexes will be discussed in the next Section 2.2.

Unlike the high-performance optimized Zen 4 cores, Zen 4c cores are optimized by die area for energy efficiency [23, 119]. Zen 4c cores are not used in all types of EPYC Zen 4 processors (see Table 5 below), and they do not add anything important to the discussion of the core microarchitecture here, other than what has been said here above. For more information on Zen 4c cores, see [119].

2.2. EPYC Zen 4 Processors Microarchitecture

The primary source of data for all further analysis in this section is [23]. Where information was obtained from other sources, appropriate references are provided.

The IPC value, which increases due to the improvements made in the cores, is only one, aggregate (for the core) indicator among the information about many components of the processor, which manufacturers primarily pay attention to, including when providing tabular data on the progress achieved in new generations of x86. Perhaps more attention, especially from AMD, has recently been paid to the increase in the number of processor cores.

Hierarchy of building EPYC processors from separate cores includes several levels, which is due to the large number of cores used and the use of multi-dies technology (chiplets).

The next level of hierarchy above the cores is the core complex, CCX. CCD (Core Complex Die) dies are built from them, and several CCD dies form a whole processor. Before we begin to analyze this hierarchy, we will

TABLE 5. Zen 4 Processors Specifications

	EPYC 8004 series (1 socket)	EPYC 9004 series (1 socket) (2 sockets — except Genoa 9004P)		
Code name	Siena	Bergamo	Genoa	Genoa 9004F/Genoa 9004X
Core process technology	TSMC 5 nm (6 nm for IOD)			
Socket	SP6	SP5	SP5	SP5
Compute cores	Zen 4c	Zen 4c	Zen 4	Zen 4
Model numbering	8004P, 8004PN	9704	9104-9604[5]	9004F/9004X
Max number of cores (threads)	64(128)	128 (256)	96 (192)	48(96)/96(192)
Max cores per Core Complex (CCX)	8	8	8	8
Max number of Core Complex Dies (CCDs)	4	8	12	8/12
Max number of CCXs per CCD	2	2	1	1
Max L3 cache size (per CCX), MB	128(16)	256 (16)	384(32)	256(32)/1152(96)
Max # of DDR5 memory channels	6	12	12	12
Max memory bandwidth, GB/s[1]	230.4	460.8	460.8	460.8
Max memory per socket, TB	3	6	6	6
Max PCI Gen 5 lanes	96	128	128	128
Max lanes CXL2 1.1+	48	64	64	64
Max Processor Frequency, GHz[3]	3.15	3.15	4.15	4.4/4.2
Max All-core boost frequency, Ghz	3.1	3.1	3.55[6]	3.95[7]/3.7[8]
Max peak performance, GFLOPS[4]	3532.8	6912	5529.6	4147.2/5875.2

Data from [22, 49, 120–122].

[1] In addition to the used DDR5-4800, according to [123], DDR5-5200 is also expected;

[2] Compute eXpress Links;

[3] Here are the boost frequency values;

[4] Calculated based on the base frequency for the upper models — EPYC 8534P, 9754, 9654 and 9474F/9684X respectively;

[5] The traditional replacement of different digits with 0 is used for model numbering;

[6] for EPYC 9654, 3.9 for EPYC 9254;

[7] for 9474F, 4.15 for EPYC 9174F;

[8] for EPYC 9684X, 4.20 for EPYC 9184X.

Note: IOD (I/O Die) is an input/output die, see below. The values listed in the table as maximum values may not be achieved simultaneously (in one specific EPYC model).

point out Table 5, which is common for this discussion and subsequent performance analyses. The features of the EPYC 4004 series processors are described below in a separate subsection. EPYC 4004 has almost all of the microarchitecture features discussed below, differing mainly in purely quantitative indicators.

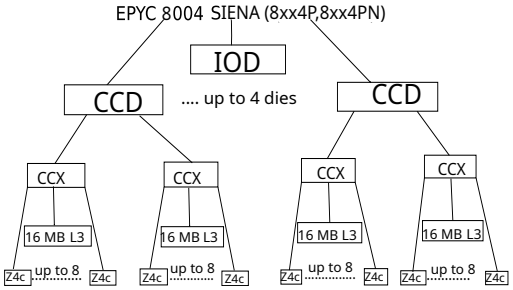


Figure 5. Siena processor design hierarchy from cores (Z4c in the figure). L3 cache is shared across the entire processor

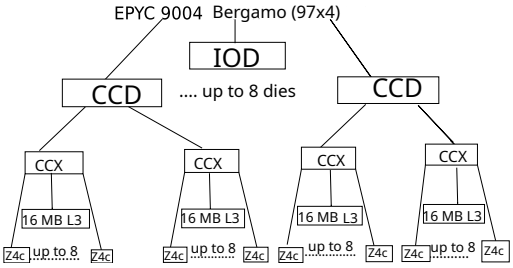


Figure 6. Bergamo processor design hierarchy from cores (Z4c in the figure). The L3 cache is shared across the entire processor

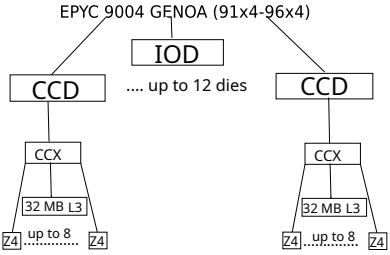


Figure 7. Genoa processor design hierarchy from cores (Z4 in the figure). L3 cache is common for the entire processor

Zen 4 includes three series of AMD server processors: 9004, 8004 (see Figures 5–7), and 4004.

The 9004 series processors use the large SP5 socket (LGA 6096 [**124**]),

which has a significantly larger number of pins for the chip than the 5th generation of Xeon Scalable processors. The 9004 series includes models that can operate in a 1P configuration only or in 1P and 2P variants. The 9004 series processors can use Zen 4 cores or Zen 4c cores.

The 8004 series processors (codenamed Siena) with a lower TDP and a simpler SP6 socket use Zen 4c cores, which contain half the L3 cache capacity of Zen 4 cores. Siena can only be used in 1P servers. The 8004 series is aimed at systems with small physical dimensions and operating in environmentally difficult conditions [**23**]. The 8004 series models come in two classes — with numbers 8004P, and with numbers 8004PN. The latter class of models supports NEBS (Network Equipment-Building System) conditions typical for US telecommunications companies [**125**].

And the 4004 series processors (codenamed Raphael) belong to the class of servers with the lowest (compared to other series) performance and certain differences in microarchitecture from the other two series. EPYC 4004 are not intended for use in the areas of HPC and AI, which are the primary focus of this review, and their features are discussed briefly below in a separate subsection.

In Siena processors, up to 8 Zen 4c cores with a shared 16 MB L3 cache are combined into a CCX, and up to two CCXs form a CCD die with a 32 MB L3 cache. Finally, up to 4 CCD dies together with an IOD die form a processor (see Figure 5). Up to 8 cores per CCX give up to 16 cores per CCD and up to 64 cores per processor. The total L3 cache capacity is accordingly up to 128 MB; it is shared by all Zen 4c cores. Other data on Siena processors are given in Table 5. Further in the review, these processors are practically not considered, since they do not correspond to its main topic.

The 9004 series processors with the Bergamo codename are also built on Zen 4c cores (see Figure 6). Bergamo uses the same hierarchy up to the CCD die, but the processor can contain up to 8 CCD dies instead of four, and accordingly up to 128 Zen 4c cores with a total L3 cache capacity of up to 256 MB. Other data, including peak performance for FP64 and used memory, are given in Table 5.

The Bergamo family of processors is aimed at scaling with high energy efficiency, and their natural application is cloud computing tasks [**22**, **107**].

The 9004 series processors with the codename Genoa are built on Zen 4 cores (see Figure 7). In Genoa, up to 8 Zen 4 cores form a CCX with a total L3 cache capacity of up to 32 MB, twice as much as in a CCX based

on Zen 4c. The CCD die therefore contains not two, but one CCX. And the entire processor can contain up to 12 CCD dies and, accordingly, up to 96 Zen 4 cores with a total L3 cache capacity of up to 384 MB.

In addition, the 9004 series includes processor models with the Genoa-X codename. They use AMD 3D V-cache technology, which was previously used in EPYC Zen 3 (Milan). When using this technology, an additional cache is placed on top of each CCD die, so that the total L3 cache capacity per CCD in Genoa-X is 96 MB. Accordingly, on a processor with 12 CCDs, the total L3 cache capacity (it is shared by all cores) is 1152 MB [**23**].

To conclude this general introduction to the hierarchical construction of the microarchitecture of the various Zen 4 processor series, we present Table 6, which contains concrete models of these processors that are of interest to our review. Here it is necessary to explain the numbering system of all EPYC Zen 4 processor models.

The processor model number consists of four decimal digits, the last of which, equal to 4, denotes the model's assignment to the Zen 4 architecture. The meaning of the first digit, 8 or 9 for the 8004 and 9004 series, was just discussed above.

Let's focus on the 9004 series models, each of which has a number of $9KL4$, where K and L are decimal digits. As K increases from 0 to 6, the number of cores (C) available in the processor increases. At K=0 C=8, at K=1 C=16, at K=2 C=24, at K=3 C=32, at K=4 C=48, at K=5 C=64, at K=6 C can range from 84 to 96.

And the increase of the L number from 1 to 8 in the model number corresponds to the increase in performance [**121**]. In addition, at the end of the number there may be an additional letter F for models with an increased clock frequency, or the letter X for Genoa-X using 3D V-cache, or the letter P for models that can only work in 1P servers. A similar numbering system is used for the 8004 series.

The reason for the formation of such a large number of different processor models is their orientation to different areas of possible application and cost indicators. This became possible due to AMD's modular approach using multi-die technology and the hierarchical construction of processors described above.

Intel also offers a huge variety of different models for different areas of application in the Xeon processors discussed below. However, for understanding SKUs, AMD's modular approach described above seems clearer and more understandable.

Table 6. EPYC Zen 4 server processor models

Model number	Codename	Number of cores	Base/Boost freq., GHz	All-core boost freq., GHz	Cache L3 size, MB	TDP, W[1]	Price[2]	Peak FP64 performance, GFLOPS[3]
9754	Bergamo	128	2.25/3.1	3.1	256	360	$11900	6912
9684X	Genoa-X	96	2.55/3.7	3.42	1152	400	$14756	5875.2
9654	Genoa	96	2.4/3.7	3.55	384	360	$11805	5529.6
9554	Genoa	64	3.1/3.75	—	256	360	$9087	4761.6
9534	Genoa	64	2.45/3.7	3.55	256	280	$8803	3763.2
9474F	Genoa	48	3.6/4.1	3.95	256	360	$6780	4147.2
9454	Genoa	48	2.75/3.8	3.65	256	290	$5225	3168
9384X	Genoa-X	32	3.1/3.9	3.5	768	320	$5529	2380.8
9374F	Genoa	32	3.85/4.3	4.1	256	320	$4850	2956.8
9354	Genoa	32	3.25/3.8	3.75	256	280	$3420	2496
9334	Genoa	32	2.7/3.9	3.85	128	210	$2990	2073.6
9274F	Genoa	24	4.05/4.3	4.1	256	320	$3060	2332.8
9174F	Genoa	16	4.1/4.4	4.15	256	320	$3850	1574.4
8024P	Siena	8	2.4/3.0	2.95	32	90	$409	460.8

Data from [122] and [125] as of 9.04.2024.

[1] default value configurable values see in [125, 126];

[2] data from [49] as of 23.04.2024;

[3] calculated by us at base frequency.

The table shows the models whose performance will be discussed further or which we consider interesting for this review.

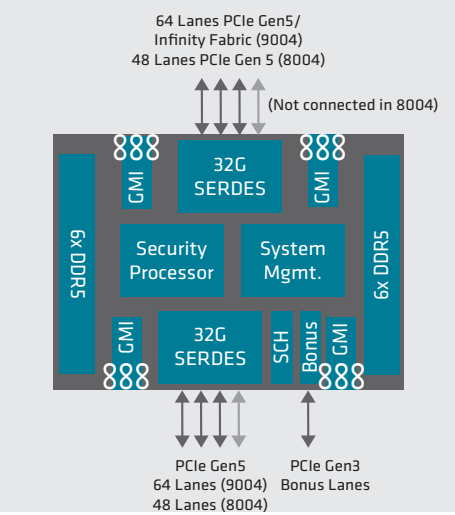


Figure 8. IOD (figure from [**23**])

IOD **die and Infinity Fabric.** All execution units that determine the processor's performance, at least relative to peak values, are located in the cores and have already been discussed above. As for the memory hierarchy, everything up to the L2 cache level also applies to the cores.

But the L3 cache, which is common to the entire processor (although the L3 blocks are located in separate processor cores), as well as the DRAM itself, are common for the entire EPYC. The Infinity Fabric interconnect (hereinafter referred to as IF in this Section 2 and in Section 3) is also used to access the memory, connecting the CCDs. The IF connects all the dies to each other [**121**]. The IF, like the memory controllers, is integrated into the IOD block in Zen 4, so the analysis should now cover this block. Although the IOD implements a very large number of different functions (which allows us to talk about Zen 4 as an SoC), it is somewhat simpler than other dies in Zen 4, and is designed using cheaper 6 nm technology. The IOD and CCD, as different dies, can be changed and improved independently.

Figure 8 provides the main components of the IOD, and Figure 9 provides a general illustration of how the IOD is used for the corresponding functions in different processor series.

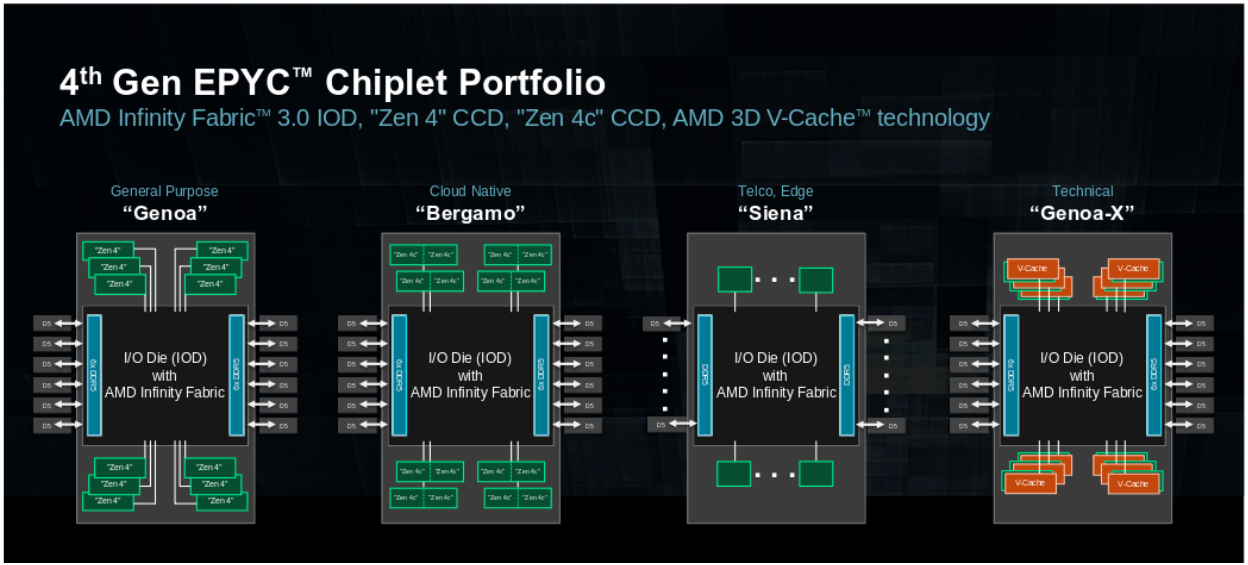


FIGURE 9. Infinity Fabric and IOD usage across different EPYC Zen 4 processor series (figure from [22])

The May 2024 version of the document [23] (the September 2023 version was mainly used in the review) also introduced information about another component in the IOD, the System Management Unit (SMU), which controls the distribution of power to the IOD, CCDs, and cores, maintaining the processor within its parameters, including those specified by power management settings. This will be discussed below when discussing TDP.

IF technology is used by AMD not only in all generations of Zen processors, but also in its GPUs [2], and is constantly being improved, including with an increase in its bandwidth. Zen 4 uses version IF 3.0, which uses lanes that operate at speeds 78% faster than IF 2.0, which was used in Zen 3 [127].

The IF topology is configurable. The base part of the IF, the Infinity Scalable Data Fabric (SDF) [128], aims to provide scalability of the interconnect. SDF is based on the earlier Hyper Transport technology, which was originally focused on high bandwidth and low latency. SDF transfers data between endpoints, such as between NUMA nodes. SDF may have connection points for PCIe PHY, memory controllers, a USB hub, and is also used to ensure cache coherency. SDF communicates with various serializers/deserializers (SerDes). The SCH block in Figure 8 is responsible for supporting interfaces for different buses — I2C, SMBus, SPI/eSPI, etc. One of the IF blocks, in particular, manages temperature and power [128].

Support of PCIe PHY at the physical level gives IF obvious advantages in flexibility. Part of the large set of SerDes lanes supported in IOD is used to work with SATA, part is used to work with CXL 1.1+, part is used to work with PCIe, and part is used to work with IF. Since IF technology is also used to interconnect processors in servers, there is the possibility of flexible configuration for concrete servers — part of the PCIe lanes at the physical level may be switched to work with IF channels [23].

The concrete configuration and application of the IOD depends on both the EPYC processor socket used, the concrete processor model, and the IOD configuration in a concrete server model. This flexibility ultimately allows for the efficient creation of a concrete server model for a specific application area. All of the following clarifications are based on [23], and additional references are provided for extensions of this information.

EPYC 9004 doubles the CPU I/O bandwidth of Zen 3 by using PCIe-v5 in the IOD. The IOD supports 12 DDR5 controllers, Infinity Fabric, SATA controllers, and Compute Express Link (CXL) 1.1+ memory. EPYC 9004 doubles the CPU I/O bandwidth of Zen 3 by using PCIe-v5 in the IOD. The IOD supports 12 DDR5 controllers, Infinity Fabric, SATA controllers, and Compute Express Link (CXL) 1.1+ memory. Their application possibilities can be configured for concrete server.

The IOD also houses the AMD Secure Processor and the memory controllers, allowing it to manage a number of memory encryption mechanisms that are part of the AMD Infinity Guard feature suite (see below for more on this in the Zen 4 security features description).

The IOD in EPYC Zen 4 has 12 IF connections to the CCDs. These CCDs can support one or two IF connections to the IOD. EPYC models with four CCDs can use two of these connections to optimize the bandwidth of each CCD. This is the case in some EPYC 9004 processors and all EPYC 8004 processors. In processor models with more than four CCDs, such as the EPYC 9004 series, one IF connects each CCD to the IOD.

EPYC processors with the larger SP5 socket have 128 PCIe lanes and 12 memory channels, while processors with the simpler SP6 socket, which has fewer contacts, support 96 PCIe lanes and 6 memory channels.

But in a 2P server configuration, part of the IOD must work for the IF connection between the processors, so a 2P EPYC 9004 configuration on both processors can only provide up to 160 PCIe lanes.

A portion of the PCIe lanes can be configured as integrated SATA controllers, up to 64 PCIe lanes in the EPYC 9004 (up to 48 lanes in the EPYC 8004) may be configured to support CXL 1.1+ for cache-coherent memory expansion and persistent memory support. In a concrete server model design, the bonus PCIe-3.0 lanes (see Figure 8) are often used to support not high performance disks (such as M.2 SSDs) used for system boot.

But more important for HPC tasks is the use of IF instead of part of the PCIe lanes. Since, as noted above, IF is compatible with PCIe PHY, at the physical level, part of the PCIe lanes can be used for IF. One IF channel uses a physical ×16 PCIe connection (this corresponds to a bandwidth of 128 GB/s for bidirectional transfer). With EPYC 9004 processors, it is possible to create 2P servers, and then up to 4 IF

channels (AMD calls them G-links, or xGMI, and what is GMI — see below) can be used for communication between processors in the server, which accordingly gives a theoretical bidirectional bandwidth of up to 512 GB/s. This is slightly more than the theoretical bandwidth of 12 DDR5-4800 channels supported in the IOD (12×4.8 GHz×8 bytes = 460.8 GB/s). Accordingly, the access speed to the memory of another processor is close to the speed of local memory [**23**].

Just as the 16 IF lanes (PCIe PHY at lower-level) are combined into a G-link, the PCIe lanes are combined into a P-link (see Figure 11 below). For interprocessor communications can be used three G-channels instead of four, and the freed-up lanes are provided as additional PCIe-v5 lanes (or can be used for CXL) [**121**] (about interprocessor communications, see Section 2.3).

There are 12 IFs, called Global Memory Interface (GMI), that can be used to communicate in the CCDs die set and with IOD. The IOD can support 4-, 8-, or 12-core CCDs with Zen 4 or Zen 4c cores. EPYC models with 4 or fewer CCDs can use by two GMIs for connection between the CCD and the IOD.

In addition, the IOD includes a security processor that provides, among other things, Secure Memory Encryption (SME) and Secure Encrypted Virtualization (SEV), a USB controller hub, and other. A brief overview of the EPYC Zen 4 security features is provided in this section below.

Although the IOD discussion started with the memory hierarchy, the main memory itself has hardly been discussed so far. We have already mentioned several times the presence of 12 channels of DDR5-4800 in the EPYC 9004, which accordingly provide a peak bandwidth of 460.8 GB/s (6 channels in the EPYC 8004 provide a bandwidth that is 2 times less). Thanks to the use of DDR5, the bandwidth in the EPYC 9004 has become twice as large as in the corresponding EPYC Zen 3 models.

For each memory channel in the IOD has its own UMC (Unified Memory Controller). Each UMC allows the use of 1 or 2 DIMM modules per channel (abbreviated 1DPC or 2DPC), and accordingly up to 24 DIMMs per slot with a maximum total capacity of 6 TB [**121**] (a typical option for creating such a capacity is the use of 3DS RDIMM with a capacity of 256 GB).

It is useful to remind here that although DDR5 supports two DIMM slots per channel, installing them on one channel causes a decrease in frequency and, accordingly, in the channel bandwidth — to combat this, as is known, buffered memory appeared, including LRDIMM (see, for example, [**129**]). Buffered memory could be used with AMD EPYC Zen 3 [**103**], and is supported by Zen 4 [**104**].

The presence of CXL 1.1+ support in IOD makes it possible to connect CXL memory as an intermediate level in the memory hierarchy between the main and external memory. This seems to be most actual for working with in-memory databases and machine learning, although CXL memory may be useful in other areas as well [**130**]. CXL memory is also expected to be effective for HPC tasks [**131**].

The somewhat late discussion of memory for Zen 4 in the review, despite its high importance for performance, was done intentionally, as there are many more subtle nuances related to NUMA memory features for EPYC Zen 4 that require a separate analysis.

NUMA memory features. When using multi-chip technology in the processor, there will be different memory latencies depending on the connection between the individual crystal (CCD in the case of EPYC) and the memory controller, i.e. there will be NUMA. The following information regarding NUMA is based on [**121**], and applies to EPYC 9004.

Thanks to the use of memory controllers located in the IOD, which was implemented in AMD Zen 2, the memory latency unevenness was significantly reduced. Subsequent increases in the L3 cache capacity in new Zen generations also in a certain sense eliminate the latency difference. The use of a new version of IF in Zen 4 compared to that used in Zen 3 also helps to reduce this difference [**23**]. However, for applications in which memory latency unevenness remains important, additional optimization is possible with explicit consideration of the possible division of the processor into NUMA domains (nodes), designated NPS (Nodes Per Socket).

As you can see in Figure 9, the AMD processors that are more important for the purposes of this review — EPYC Genoa and Genoa-X, as well as the Bergamo processors aimed at high-density computing systems, have 4 "groups" of CCDs, which correspond to 4 quadrants in the processor. From these, 4 NPS nodes can be formed (see Figure 10).

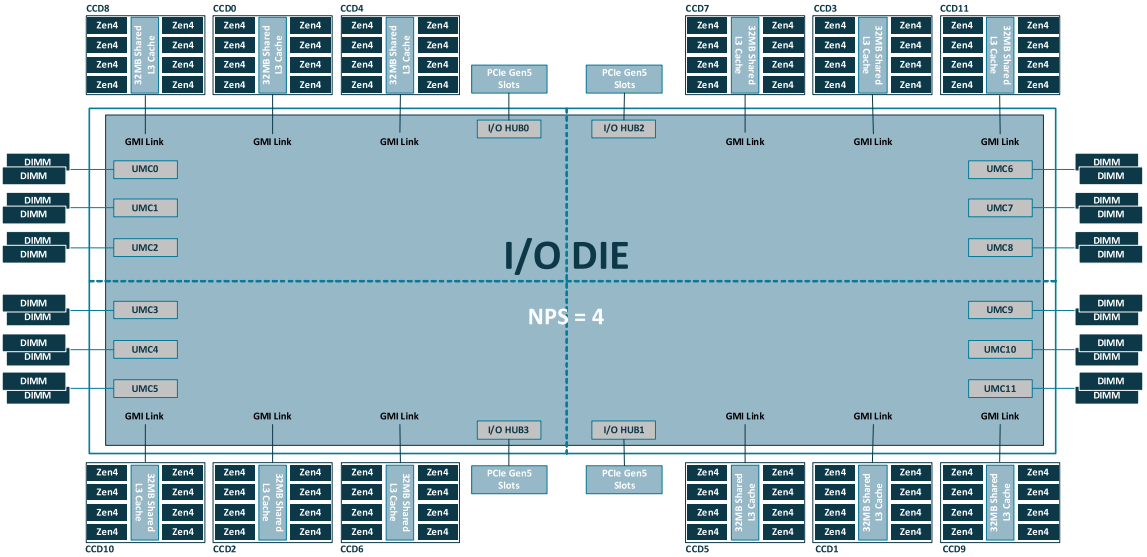


Figure 10. EPYC 9004 Genoa/Genoa-X — 4 quadrants and 4 NPS (figure from [121])

At NPS=4 the processor is divided into 4 NPS nodes, at NPS=2 — into 2 nodes, at NPS=1 the processor has one common NUMA node per socket. NPS values are set in BIOS. NUMA nodes provide localization of cores, memory, hubs and I/O devices. It is also possible to use localization of L3 cache in each CCD, and then one processor with 12 CCDs can have up to 12 NUMA nodes (for more details, see [**121**]).

It is important to note here the interleaving of memory channels, which is used to enable simultaneous operation of several memory channels by dividing the memory address space into blocks. At NPS=1, all processor memory channels are interleaved; at NPS=2, 6 memory channels are interleaved in each node; at NPS=4, interleaving is performed in each quadrant. NUMA in 2P servers is discussed further in Section 2.3 on Zen 4-based computing systems.

Obtaining information about the NUMA configuration of the system and binding the executable program to a processor core or NUMA node is possible in Linux using the numactl command.

As for the above-mentioned CXL memory (meaning not persistent memory, but supporting normal load/store operations), it can be perceived simply as a separate NUMA memory domain, increasing the NPS value by one (simply as another NUMA node without cores) — see, for example, [**130**].

TDP and frequency control in EPYC 9004. For HPC and AI tasks, which is what the review is mostly focused on, it is the processors of this series that are relevant.

The usual "default" Thermal Design Power (TDP) values used by EPYC Zen 4 models are listed above in Table 6. The BIOS also supports a configurable TDP (cTDP) mode, the range of possible values of which is known for each specific Zen 4 model. CTDP can be used, for example, by server manufacturers to optimize for their respective target settings.

For example, the 96-core AMD EPYC 9654 has a TDP of 360 W, while its cTDP is 320–400 W. For HPC workloads, cTDP is often set to a maximum of 400 W to achieve maximum performance.

EPYC has an important determinism mode in the BIOS, which can be set to Power or Performance. This mode uses the aforementioned SMU to determine the effective frequency of each core. The SMU monitors power, current consumption, and temperature in real time across all parts of the processor, and uses cTDP information [**121**].

The Performance selection reduces potential performance differences across processors in a cluster by using a common baseline reference setting for all identical processor models. The Power selection takes advantage of potential runtime variability across individual processors and can provide improved runtime performance. This allows each processor to consume power according to its individual capabilities while not exceeding the cTDP power limit. The Power selection is used to set cTDP to its maximum value [**121**].

When combined with BIOS settings BOOST=ON and Power determinism, this allows the SMU to use all available power, and can give higher effective frequencies above base and for heavy SIMD workloads (e.g. DGEMM) [**121**].

Information on the supported frequency-adjustable processor states and CPUfreq controllers in EPYC 9004 is available in [**121**], which forms the basis for the discussion below. The entire processor or its individual cores support states C0, C1, C2.

For high-performance, low-latency networks (e.g., Infiniband), as described in [**121**], the C2 state in Linux should be disabled. This type of adjustment is done using the cpupower command for individual processor cores or the entire processor. When `SMT` is enabled, such adjustments using cpupower are per "logical processors", and C2 should be disabled for the corresponding pair of logical processors.

In EPYC 9004, the C0 running state also supports P0, P1, P2 states with different frequencies. In boost mode (it can be enabled in BIOS, and then enabled/disabled in Linux), each core can operate at a frequency higher than that determined by the P0 state.

EPYC processors support 4 CPUfreq governors. For HPC, the performance governor is typically used, which selects the frequency being installed in P0. When boost mode is enabled, the governor attempts to raise the frequency to the maximum boost value [**121**].

The ondemand regulator sets the frequency based on the load being used. This allows for a rapid ramp-up to the maximum operating frequency, followed by a slow step down to P2 when idle. This is bad for short-lived threads [**121**].

The conservative regulator is similar to ondemand, but offers a smoother transition to maximum frequency and a quick return to P2 at idle. Finally, powersave sets the lowest supported frequency, locking it to P2.

The choice of the used governor is also done in Linux using cpupower.

Hardware support of virtualization. As already mentioned above, these features of Zen 4 will be discussed here very briefly. AMD-V virtualization technology has been used in the company's processors for a very long time, and the corresponding part of ISA is used for it. AMD-V also includes such functions as input-output virtualization (AMD-Vi), which supports direct allocation of devices to virtual machines, and interrupt virtualization AVIC (Advanced Virtual Interrupt Controller).

But the most important thing here seems to be minimizing the overhead of virtualization of virtual memory. AMD uses nested page tables for this, using second-level address translation (SLAT), which allows avoiding the overhead of using shadow pages. AMD has long been using its Rapid Virtualization Indexing (RVI) technology for this.

Taking into account the possibility of memory expansion using CXL controllers, 57 bits are already used for the virtual memory address in Zen 4, and the fifth level of nested page tables is implemented [**23**] (the fifth level of the page table for working with 57-bit addresses was proposed by Intel in 2017 [**132**]).

When using virtualization, security requirements increase dramatically, since it is necessary to ensure data isolation in different virtual machines. For the frequently used second level of nested pages, SNP, AMD has long used the SEV-SNP (SEV — Secure Encripted Virtualization) security features, see below.

Transparent Huge Pages (THP), used in Linux to reduce TLB misses for large memory machines, can be as large as 2 MB in Zen 4 [**121**]. THP is called transparent because applications are not explicitly notified when it is enabled in the Linux kernel, with the expectation that it will improve their performance by reducing the memory management overhead of using

the TLB by reducing the number of TLB entries required. Enabling THP in the Linux kernel can improve the performance of HPC applications, but it often reduces performance in database applications, and THP is therefore is usually disabled there.

Security. AMD has placed a strong emphasis on security in Zen 4. Naturally, these security features have been gradually developed since the first generation of Zen. AMD takes steps to address vulnerabilities known during development, and this does not require modification of application programs [**23**].

The increased use of virtualization, which is natural for multi-core Zen 4, as noted above, requires increased isolation of virtual machines, and was accompanied by a corresponding increase in hardware support for security tools. Since AMD has a number of such tools, some of which have their own descriptions, we will limit ourselves here mainly to providing links to them. These tools are Infinity Guard (it integrates the above-mentioned SME and SEV-SNP tools) [**133**], memory security features [**134**], shadow stack, and side channels protection.

Most of the potential security issues are present in all modern processors. The shadow stack and side-channel protection are directly related to virtualization problems [**135**]: strong isolation between different virtualized guests is required, and each virtual machine needs its own encryption key. Secure nested paging is also provided. The `IOD` includes its own security processor; multi-key encryption (SMKE) is also supported [**133**], which allows hypervisors to selectively encrypt ranges of address space in CXL memory [**23**].

Publications on the security features of Zen 4 are likely to appear soon. Here we point to the work [**136**] that analyzes the trusted platform modules and the capabilities of the AMD security processor in Zen 3. And currently security studies are also being conducted for AMD Zen 2 [**137**, **138**].

Peculiarities of EPYC 4004 series processors. These processors are used in servers with a small number of cores and a single socket. Information about EPYC 4004 is available in the May 2024 version of the document [**23**], which is the basis for the brief consideration in this subsection.

EPYC 4004 contain from 4 to 16 cores and up to two CCDs. The maximum possible L3 cache capacity is 64 MB, and 3D V-cache is 128 MB. These processors use a simpler IOD with a smaller area compared to series for high-end processors.

This IOD has a number of modifications compared to other EPYC Zen 4 series. In particular, an AMD GPU with RDNA2 architecture has been added. The IOD has one SerDes device with 16 lanes and one with 12 lanes, which gives up to 28 PCIe-v5 lanes. There are 2 channels of DDR5-5200 memory with a total theoretical bandwidth of 83.2 GB/s, and the capacity of this memory can reach 192 GB.

But by using a small number of cores, they can operate in the processor at high clock rates, increasing the performance of each core. Thus, the 16-core EPYC 4564P model has a base frequency of 4.5 GHz, and an boost frequency of up to 5.7 GHz [**139**].

The security processor in the IOD of this EPYC series does not support SEV and SMKE. EPYC 4004 is targeted at small business servers, dedicated server hosting, and other applications where low core count processors are sufficient.

2.3. About EPYC 9004-based Computing Systems

The EPYC 8004 and 4004 processors are only available in single-socket computers and are not discussed further. In reality, this section could cover everything from motherboards to supercomputers, since only processors were analyzed previously. However, in accordance with the focus of the review, the discussion below is limited to some features related to the EPYC 9004. All leading manufacturers produce corresponding motherboards and servers. And there are a wide variety of server and multi-module server systems with both air and liquid cooling, both without and with GPUs.

EPYC 9004 can be used to create 1P and 2P servers. In the latter, as noted above, 3 or 4 IF links (G-links, see Figure 11) are used for communication between processors. For servers that require intensive input/output, 3 G-links can be used, then the freed link can be allocated for PCIe-v5, obtaining a total of 160 PCIe lanes for the entire server.

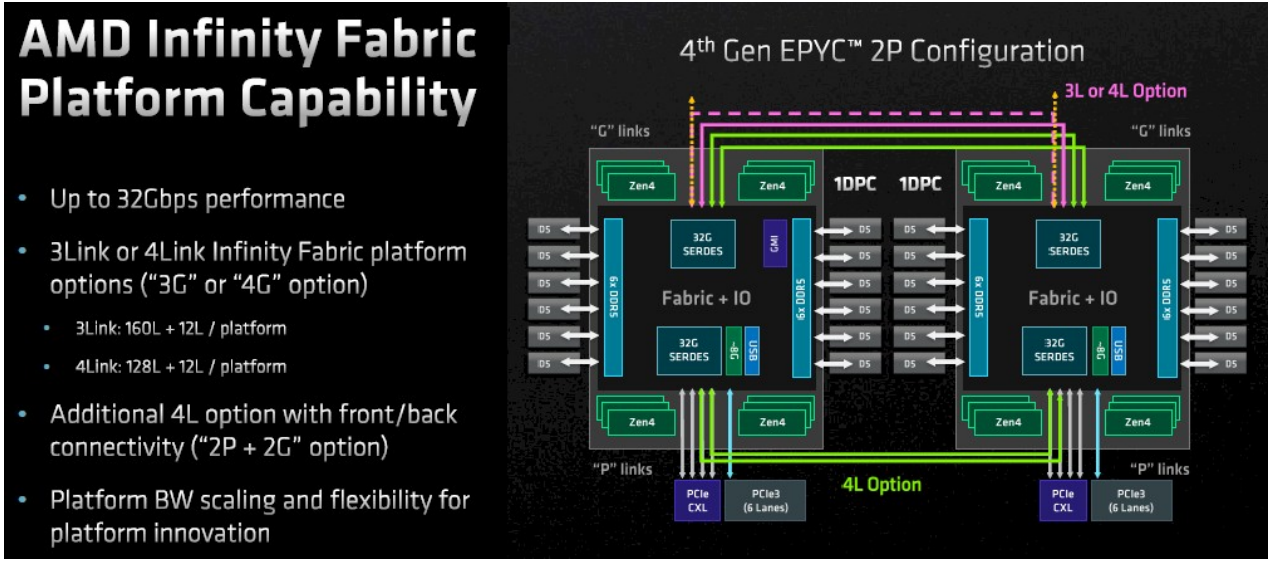


Figure 11. 2P configuration with EPYC 9004 (figure from [127])

AMD's focusing on HPC is evident even in the BIOS manual. BIOS settings can be viewed in Linux, and some can even be changed. The BIOS of servers with the EPYC 9004 has a number of interesting options that can potentially improve performance or energy efficiency. The number of cores used in the CCD can be reduced in increments of 1 from 8 to 1, while keeping the total L3 cache size constant. The number of active CCDs in the processor can be reduced, while keeping the number of cores per CCD the same [**121**].

As already mentioned in Section 2.2, there is an option in BIOS that allows or prohibits increasing the used processor frequency. If it is enabled, it can be disabled in the Linux command line. It is clear that there is an option to enable/disable SMT (each core can support one or two threads). There is an option that increases the efficiency of interrupt processing in Linux in a configuration with a large number of processor cores.

The memory speed can be explicitly set (up to 5400 million transfers per second, which means Zen4 can also work with DDR5-5400). A dual-socket system has 24 memory channels. Setting NPS=0 in the BIOS suppresses the use of NUMA, and all shared memory connected to both server processors appears as homogeneous. The memory is herewith interleaved across all channels in a single server address space. Memory interleaving can be disabled in the BIOS [**121**].

A number of options relate to xGMI. For more information on xGMI and BIOS options, as well as other important for HPC features including Mellanox interconnects, see [**121**].

Similar to the guide [**121**] for HPC, AMD offers a guide for efficient AI work [**140**], which describes, for example, how to efficiently create Hadoop clusters focused on big data processing.

In terms of security, EPYC 9004-based servers receive FIPS 140-3 certification (e.g. Lenovo server [**141**]).

In conclusion of the section on computing systems with EPYC 9004, we will illustrate them with examples of servers with these processors, and nodes of TOP500 supercomputers using them.

As a typical option for constructing such a server, we can point to the detailed block diagram for the Lenovo ThinkSystem SR665 V3, which is a traditional 2P server with a 2U form factor [**142**] (see Figure 12).

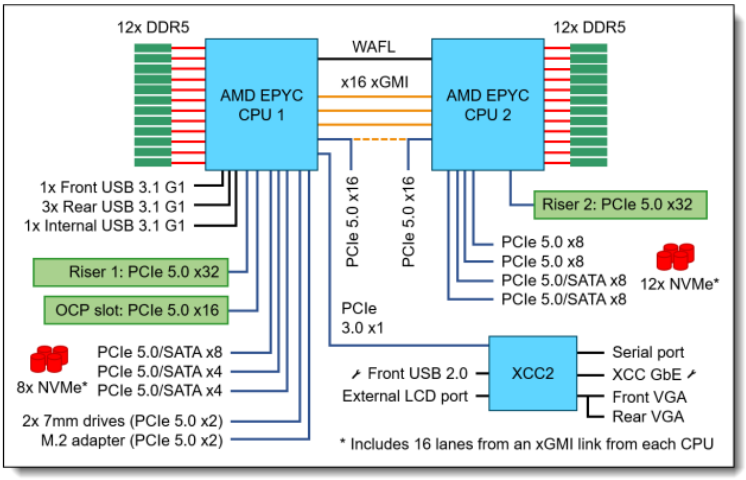


Figure 12. Lenovo SR665 V3 server architecture block diagram (figure from [142])

An example of supercomputer with homogeneous nodes is Shaheen III in Saudi Arabia, which was launched in 2023 and ranked 23rd in the June 2024 TOP500 list. The nodes there contain by two EPYC 9654 processors [143].

As an example of heterogeneous nodes with GPUs, we cannot help but point to the world's first exascale supercomputer Frontier. Although the nodes there use 64-core EPYC 7A53 with Milan cores (one processor per node), the processor already uses a more modern IOD [144]. About the leader in the TOP500 list since November 2024 supercomputer El Capitan, whose heterogeneous nodes use Zen 4 processors, is said in Section 7.

Another interesting example of a supercomputer from the June (2024) TOP500 list with an Nvidia H100 GPU is the German HoreKa-Teal supercomputer, which ranks 6th in the corresponding GREEN 500 list. HoreKa-Teal nodes use Lenovo ThinkSystem SD665-N V3 servers, each containing two 32-core EPYC 9354 [145].

2.4. About the Performance of Computing Systems with EPYC Zen 4

This article will focus on performance data for servers with Zen 4 (EPYC 9004) processors, including comparisons to previous generation x86

server processors and ARM processors, although in some cases a useful comparison to GPUs is also included. Cluster performance data is provided solely to demonstrate how specific applications scale in clusters.

When considering the comparison of the performance of different processors in the review, we primarily pay attention to the comparison of high-end models (which are often used in supercomputers), as well as to the comparison of models with the same number of processor cores. Separately, we also pay some attention to the middle-class models, which are actively used in HPC, and here the most interesting is the comparison of models with the same number of cores.

It is clear that it is also interesting to compare with similarly priced models or with similar TDP values, but this is considered somewhat secondary here. In addition, even a comparison of the performance of models with different numbers of processor cores is of interest, since this can give very rough initial assumptions about the possible scaling of performance with the number of cores.

As for the performance increase of EPYC Zen 4 compared to Zen 3, it is obvious due to the increase in the number of cores, their clock frequencies, the appearance of AVX-512 support, the increase in the bandwidth of the used memory, the capacity of the 3D L3 cache, etc. It is clear that this also gives an increase in the performance of Zen 4 models with the same number of cores as Zen 3.

There is a wealth of data on EPYC Zen 4 performance: on well-known mass benchmark sites spec.org, openbenchmarking.org, on AMD sites (most notably on the documentation hub [**146**]), or for example on other sites focused on specific application areas. They show Zen 4 performance improvements over Zen 3, and usually performance advantages over Xeon `SPR` and `EMR`.

The review presents only a small part of such data, primarily related to HPC, and selected as more actual. As a general illustration of the performance acceleration of known HPC applications in the senior Zen 4 Genoa model (EPYC 9654) compared to the senior Zen 3 model (EPYC 7763), we will point to the data from the AMD report at the seminar in Academia Sinica (National Academy of the Republic of China Taiwan) [**89**], see Figure 13.

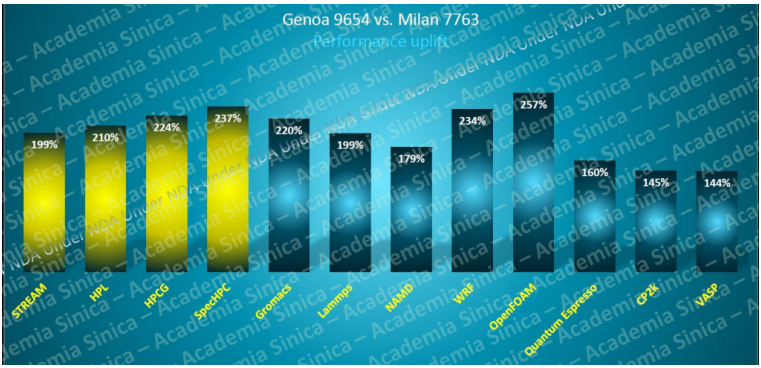


Figure 13. Relative performance of EPYC 9654 compared to EPYC 7763 on benchmarks and applications (figure from [89])

However, it can be assumed that these numerical data (in percent) on the performance increase are not average, and it would be more reasonable to mark them with the "up to" sign. Many performance data on the various Zen 3 and Zen 4 models for the applications and benchmarks shown in this figure, as well as many others data, can be found by searching the openbenchmarking site for the benchmark name, which will yield a URL like https://openbenchmarking.org/test/pts/cp2k-1.3.0, where cp2k is a possible application (or benchmark) name, followed by a dash, and the version number.

Performance data for 1P and 2P server configurations with EPYC 9654 and 9554 models on a wide range of different applications, including those showing performance advantages over EPYC Zen 3 and 3rd Gen Xeon Scalable processors, is available in [147]. Performance comparisons of EPYC 9004 to Xeon EMR and Xeon SPR/Xeon Max are primarily provided later in Section 4.1–Section 4.4.

As another example covering performance data for a wide range of different application areas, including many not covered in this review, we will point to [148]. AMD EPYC has become very actively used for PTS benchmarks, where for a 2P server with EPYC 9654, even comparisons of the achieved performance were made when using different modern Linux distributions without any modifications on a wide range of benchmarks, including molecular dynamics (NAMD), database work (PostgreSQL and MariaDB), AI tasks (including OpenVINO and OneDNN benchmarks), and many others [149].

AMD provides a lot of performance data for the EPYC 9004, and we have selected only the most actual ones. But before discussing specific benchmark data, including application benchmark data, it is useful to point out the general information about the speedup achieved by using AVX-512 support in Zen 4, which was obtained in [**150**] when testing a 2P server with an EPYC 9654. It demonstrated significant speedups on many AI benchmarks, including TensorFlow (with AlexNet, ResNet-50, and GoogleNet models), OpenVINO, some oneDNN benchmarks, etc. The EPYC 9004 speedup in AI benchmarks was expected due to the support of the AVX-512 extension implemented in Zen 4 for VNNI capabilities and the BF16 format.

A noticeable performance increase using AVX-512 was obtained in [**150**] in the molecular docking miniBUDE mini-application. This may be due to the use of 3D matrix transformations there [**151**]. However, in other HPC applications that do not use traditional dense matrix multiplications, including the CP2K molecular dynamics application or the OpenFOAM CFD tools, the performance gain from enabling AVX-512 support in [**150**] was small.

2.4.1. *Performance Data for Various SPEC Benchmarks*

It is natural to start the analysis with data from various SPECcpu 2017 benchmark. The corresponding results for floating-point performance are given below in Table 27 (it provides data for this benchmark not only for EPYC 9004, but also for Xeon SPR/Xeon Max and Xeon EMR). A lot of data showing the performance advantage in SPECcpu 2017, SPEComp 2012, SPECmpi 2007 and SPEChpc 2021 benchmarks of various high-end and mid-range EPYC 9004 models compared to the corresponding EPYC Zen 3 and Xeon ICL (third generation scalable processors) models, including when compared with the same number of processor cores, is given in the AMD paper for HPC [**152**].

Because the performance advantages of the EPYC 9004 over EPYC Zen 3 processors are fairly clear, our analysis of the SPECcpu 2017 benchmark data focuses exclusively on performance comparisons with Xeon SPR, EMR, and Xeon Max, and is carried out further in the separate Section 4.1 and Section 4.4. For similar reasons, the SPEChpc 2021 benchmark data in Table 28 is also included in Section 4.1, which primarily focuses on performance comparisons of the EPYC 9004 to 4th and 5th Gen Xeon Scalable processors.

Table 7. Performance data of the considered x86 processors in the SPEComp 2012 benchmark

SKU	CPUs	Cores	SPEComp_2012 result	
			Peak	Base
EPYC 9654	1	96	48.4[1]	47.5[1]
	2	192	86.0[1]	83.0[1]
EPYC 9754	1	128	64.2[2]	61.7[2]
	2	256	108[2]	104[2]
Xeon 8480+	2	112	—	61.0[3]
Xeon 8490H	2	120	—	61.3[3]
	4	240	108[4]	100[4]
	8	480	138[5]	126[5]
Xeon 8592+	2	128	—	62.0

The maximum achieved data is given as of 1.09.2024.

The senders of the result and the servers used:
[1] Lenovo ThinkSystem SR655 V3;
[2] Cisco UCS C245 M8 SM;
[3] xFusion FusionServer 2288H V7;
[4] Lenovo ThinkSystem SR860 V3;
[5] Lenovo ThinkSystem SR950V3.

As for the SPEComp 2012 benchmarks, the corresponding results from [**153**] are given in Table 7.

In all the servers presented in the table, on which benchmarks were performed, the number of processor cores differs quite significantly from each other, and higher performance is simply associated with a larger number of cores.

The data in this table used the SMT capabilities of the processors, and used 2 threads per physical core. The data in this table, during the transition from 1P to 2P servers, and from 2P to 4P servers, shows how the performance of these benchmarks can scale as the number of cores and, correspondingly, the number of OpenMP threads used increases.

The performance of a single 128-core EPYC 9754 (Bergamo) is only slightly higher than the 2P server with Xeon 8480+ (with 112 cores per

server) for the base (without individual optimization for each benchmark component) result, and slightly lower than the 2P server with Xeon 8592+ (with 128 cores per server). But it should be borne in mind that Bergamo processors are not initially focused on the HPC area and are based on slower cores compared to EPYC Genoa, and the 2P server with EPYC Genoa (with 96-core EPYC 9654 models), naturally, outperforms the 2P servers with Xeon in this benchmark.

The second important point here is memory bandwidth. When the total number of all cores in the compared servers is close, the 2P system gains an advantage, since each processor has its own memory controllers, and the total memory bandwidth doubles. Therefore, the fact that the performance of a server with one 128-core Bergamo EPYC 9754 processor is very slightly inferior to a 2P server with two top 5th generation Xeon 8592+ processors with the same total number of 128 cores, speaks of a combination of high competitiveness in terms of performance of individual Bergamo cores and its good scaling (and Genoa — even more so).

The SPEComp 2012 is an integrated for HPC benchmark (its components cover different application areas), and so the results are highly actual. However, it would be useful to examine how the performance scales with the number of cores used in the calculation.

[154] provides interesting data showing the impact of the number of threads used on the SPEComp result when testing the performance of the EPYC 9654 with SMT support enabled. The performance increases by only 25% when moving from one core (2 threads) to two cores (4 threads), but continues to increase up to the use of all 96 cores (192 threads). On 96 cores, the achieved speedup was more than 24 times. But on the other hand, this does not correspond to the known data on possible performance scaling stops with an increase to a much smaller number of cores, including in CFD applications, and general typical recommendations for choosing processors for CFD tasks [**155**]. At the same time, 5 out of 18 benchmarks in SPEComp relate to CFD, not counting the aerodynamic benxhmarks on weather forecasting (these areas are memory-bound). Such situations with a rapid decrease in performance growth with an increase in the number of used cores often occur in other HPC applications, see below.

Table 8. Performance of some AMD EPYC models in the SPECmpiM 2007 benchmark

Model	CPUs	Number of Ranks	Compute Threads Enabled	Cores	Base	Peak
EPYC 9754[1]	1	128	256	128	36.4	36.4
EPYC 9654P[2]	1	96	96	96	36.4	36.4
EPYC 9654[3]	1	96	192	96	36.0	36.0
EPYC 9654[4]	2	192	384	192	64.1	64.1
EPYC 9654[5]	2	192	192	96	65.4	65.4
EPYC 9755[6]	2	256	256	256	96.6	96.6

Data senders and servers on which the benchmark was performed:
[1] Supermicro A+ Server 1025CS-TNR
[2] Lenovo ThinkSystem SR655 V3
[3] Supermicro A+ Server 1115CS-TNR
[4] Supermicro A+ Server 2125HS-TNR
[5] Lenovo ThinkSystem SR665 V3
[6] Supermicro Hyper A+ Server AS -2126HS-TN. Data from [156] as is 5.12.2024.

It is also worth noting here that Lenovo [154], whose servers produced some of the highest SPEComp results for EPYC 9004 listed in this table, shows that enabling SMT mode increases performance by more than 1.5 times. This contradicts the known data on the weak impact of enabling SMT on performance in HPC applications and, in particular, the recommendation to disable SMT for CFD applications [155]. But it is clear that studying the dependence of performance on the number of cores used is very important.

Table 8 shows the performance data for the EPYC 9004 (all data refers to one cluster node) in the SPECmpiM 2007benchmarks. It also shows data for the EPYC 9755 processor (Zen 5, see about this Section 6).

There are no corresponding data for the Xeon SPR, Xeon EMR, and Xeon Max processors discussed in this review, and the 2P server with 56-core 3rd-generation Xeon (8380H) showed a result of 40.4 (the same base and peak values), which is slightly higher than the 1P server with the EPYC 9654.

The EPYC 9004 performance characteristics for Java applications and correspondings server workloads in SPECjbb2015 are shown in Table 9.

TABLE 9. EPYC and Xeon performance data in SPECjbb2015 benchmarks

Model	CPUs	composite		MultiJVM		Distributed	
		max_jOPS	critical_jOPS	max_jOPS	critical_jOPS	max_jOPS	critical_jOPS
EPYC 9754	1P	405933	379972[1]	**446246**	192284	**441549**	195109
				362000	**335166**[1]	366259	**349986**[1]
	2P	**645699**	538879	**892492**	401899[11]	**888595**	320876
		632433	**578007**[2]	703744	**648506**[2]	706963	**659783**[2]
EPYC 9684X	1P	**442094**	**414785**[1]	**494311**	218746	**497170**	222613
				383203	**357304**[1]	388313	**357576**[1]
	2P	**628244**	566406	**988621**	410550[2]	**970665**	387180[2]
		621185	**571538**[3]	741034	**715483**[12]	745969	**717538**[12]
EPYC 9654	1P	**381145**	356013[1]	494311	218746[1]	**424600**	190451
		376911	**356425**[4]			344964	**327561**[1]
	2P	607067	556933[5]	**967587**	373830[13]	**838821**	361028
				741034	**715483**[12]	672893	**635876**[2]
Xeon 8592+	1P	—	—	258368	152750[14]	—	—
	2P	-	-	558626	310911[15]	-	-
				480007	421207[8]		
Xeon 8580	2P	391485	155366[6]	464765	254278[6]	—	—
Xeon 8570	2P	372676	147612[6]	451099	233491[6]	—	—
Xeon 8558U	1P	—	—	203534	112446[16]	—	—
Xeon 8490H	1P	200721	124502[7]	213001	125650[17]	—	—
	2P	421768	261161[8]	505379	257205[18]	-	-
				458295	368979[8]		
	4P	428829	299584[9]	884811	472868[19]	**814136**	322485
				733918	618248[9]	743435	**624702**[9]
	8P	—	—	1356786	1012835[20]	1127931	896441[20]
Xeon 8480+	2P	343031	309274	**476987**	243526[21]	**421268**	213428[22]
		338796	**309284**[10]	371890	**313390**[10]	373578	**309482**[10]
Xeon 8470	2P	**334561**	226875	**402335**	210668	**402335**	207396
		317651	**297873**[10]	363716	**299820**[10]	359630	**298159**[10]

Data from [https://spec.org/jbb2015/results/ Accessed 29.08.2024] The maximum results obtained as of 29.08.2024 are shown. The senders of these results and the systems used:
[1] ASUS RS520A-E12-RS12U; [2] ASUS RS720A-E12-RS12; [3] Dell PowerEdge R7625;
[4] Lenovo ThinkSystem SR655 V3; [5] ASUS RS700A-E12-RS12U; [6] Nettrix R620 G50;
[7] Supermicro SuperServer SYS-521E-WR; [8] ASUS RS720-E11-RS12U;
[9] Lenovo ThinkSystem SR860 V3; [10] Dell PowerEdge MX760c;
[11] Kaytus KR1280V2(KR1280-E2-A0-R0-00); [12] Lenovo ThinkSystem SR665 V3;
[13] Cisco UCS C245 M8; [14] Supermicro motherboard X13SEI-TF; [15] H3C UniServer R4900 G6;
[16] Supermicro SuperServer SYS-521C-NR; [17] IEIT I22G7; [18] xFusion FusionServer 5288 V7;
[19] xFusion FusionServer 5885H V7; [20] Lenovo ThinkSystem SR950 V3;
[21] Inspur NF5180M7; [22] Dell PowerEdge R760.
All results in any cell are for tests of a single server.

The data in this table only includes those processors that may be of greatest interest to the author. Since the maximum average critical_jOPS can be achieved in one benchmark execution, and the highest max_jOPS in another, in many cases the maximum performance using the same processor is reflected by the pair of results presented in this table. Although the top row of such a pair always contains the data with the maximum max_jOPS, for convenience the maximum result of the max_jOPS, critical_jOPS from pair is marked in bold.

Interestingly, in all three variants of the SPECjbb 2015 benchmark, systems with 128-core Bergamo EPYC 9754 with reduced cache sizes and core frequencies were inferior to systems with 96-core Genoa EPYC 9684X (2P systems were inferior only in two of the three variants), but outperformed systems with 96-core Genoa EPYC 9654 that do not have an extended L3 cache (except for the MultiJVM variant).

The data from these benchmarks for the EPYC 9004 is also interesting for evaluating performance with Bergamo processors and for assessing the impact of the expanded L3 cache on performance.

The energy efficiency (performance per Watt) benchmark data, SPECpower_ssj 2008 (see Table 10), relate to Java server side business applications. Interestingly, the various ARM processors (from Ampere) do not show the expected success for ARM in these tests, and lag behind x86 (the data for Ampere Altra Q64-30, not presented in the table, gave an even lower result).

The results of this table demonstrate a strong advantage of the high-end EPYC 9004 models over the high-end Xeon models of the 4th and 5th generations. Multiple "world records" for SPECpower_ssj 2008 belong to various servers based on the energy-efficient Bergamo processors (the EPYC 9754 top model). The maximum results in these benchmarks were achieved on 2P servers. A further increase in the number of Xeon SPR processors in a server (there can be up to 8 of them) leads to a decrease in energy efficiency.

Below in Table 11 are data that are already quite far removed from the evaluation of the processor's performance, but are integral indicators for widely used data centers. These are the SPECvirt_sc2013 consolidated virtual server performance benchmarks, which use mixed workloads:

TABLE 10. Performance per Watt data for different processors in SPECpower_ssj 2008 benchmark

SKU	CPUs	Cores	Overall ssj_ops per Watt	Price[14]
EPYC 9654	1	96	27448[1]	$11805
	2	192	30602[2]	
EPYC 9754	1	128	36637[3]	$11900
	2	256	37678[1]	
Ampere Altra Q80-30	1	80	12718[4]	$3950[15]
Ampere Altra Max M128-30	1	128	11497[5]	$5800[15]
	2	256	12195[6]	
Xeon 8592+	1	64	17224[7]	$11600
	2	128	20408[8]	
Xeon 8490H	1	60	14537[9]	$17000
	2	120	17415[10]	
	4	240	15078[12]	
	8	480	11898[12]	
Xeon 8480+	1	56	10290[13]	$10710
	2	112	16653[10]	

The table shows the maximum indicators achieved as of 15.11.2024 for the listed processors from [https://spec.org/power_ssj2008/results/ Accessed 15.11.2024]. Hardware manufacturers and servers used:
[1] Lenovo Think System SR655 V3;
[2] ASUS RS720A-E12-RS12;
[3] Lenovo Think System SD535 V3;
[4] FALINUX AnyStor-700EC-NM;
[5] Supermicro, GIGABYTE R152-P31;
[6] FOXCONN (FII), Mt. Collins;
[7] Lenovo ThinkSystem SD530 V3;
[9] Supermicro SuperServer SYS-521E-WR;
[8] ASUS SC4000-E11;
[10] xFusion FusionServer 2288H V7;
[11] Lenovo ThinkSystem SR860 V3;
[12] FUJITSU Server PRIMERGY RX8770 M7;
[13] Supermicro SuperServer SYS-521C-NR;
[14] Price data from [49] as of 15.11.2024, excluding prices for ARM processors;
[15] Data from [157].

Web servers, mail servers, database work, and other widely used types of workloads.

Although there is not as much comparative data for these benchmarks as for other SPEC benchmarks used in the review, and the results depend

Table 11. Processor performance data in SPECvirt benchmarks

Model	CPUs	SPECvirt_sc2013VMs	Platform
EPYC 9654	2P	8336462	China Mobile (Suzhou) Software Technology
Xeon 8592+	2P	9801546	xFusion Digital Technologies Co., Ltd.
Xeon 8490H	2P	7528420	xFusion Digital Technologies Co., Ltd
Xeon 8480+	2P	5277294	China Electronics Cloud Technology Co., Ltd.
Xeon 8280L	8P	11966672	Lenovo Global Technology

Data from [https://spec.org/virt_sc2013/results/specvirt_sc2013_perf.html Accessed 11.02.2025]. Note: The number of virtual machines is given after . The maximum achieved values are given for servers with the given processor models as of 29.08.2024.

significantly on different indicators in both hardware (e.g. memory capacity) and software (e.g. hypervisor used), the obtained performance data generally corresponds to expectations. One can only note the success of the Xeon 8592+. This data can also be useful for assessing virtualization capabilities in general.

It can be said that in all the different SPEC benchmarks listed, the high-end EPYC 9004 processors outperform the Xeon SPR and EMR (with the exception of some data for SPECvirt_sc2013).

2.4.2. *Stream Memory Bandwidth Benchmarks*

We begin analyzing the performance of the EPYC 9004 in STREAM benchmarks immediately after considering the SPEC benchmarks due to the increasing number of situations where applications are memory-bound. The increase in memory bandwidth when upgrading from EPYC Zen 3 to Zen 4 is obvious simply because of the transition from using DDR4-3200 in Zen 3 Milan to DDR5-4800 in EPYC Zen 4. The bandwidth in the STREAM Triad doubles when replacing a 2P server with 64-core EPYC 7773X or EPYC 7763 with a 2P server with 96-core EPYC 9654 [88, 158].

In STREAM benchmarks, bandwidth may depend (in addition to the array size) on a number of additional parameters: the number of threads in OpenMP, the NUMA configuration used (NPS parameter), the use of SMT

TABLE 12. Optimal STREAM parameters for EPYC 9004 models

SKU	Number of CCDs	Optimal memory size of each array
EPYC 9654	8	2.1 GiB
	12	3.2 GiB
EPYC 9754	8	2.1 GiB
EPYC 9684X	8	6.4 GiB
	12	9.6 GiB

mode, enabling/disabling the processor boost mode, using 3D V-cache, and the number and type of DIMMs used. Most of the results discussed below provide data for the most widely used benchmark of assigning a linear combination of two vectors to a third (STREAM Triad). If another version of the STREAM benchmark is used, this is indicated explicitly.

The necessary baseline data for this benchmark, with dependencies on the above parameters, is provided in AMD's EPYC 9004 HPC Tuning Guide [**121**]. The corresponding results are presented there for 2P servers using 1DPC and dual-rank DIMMs (which gives the maximum bandwidth) DDR5-4800 with EPYC 9654, 9754, and 9684X models, using 8 or 12 CCDs in these processors. This document also provides data with comparisons regarding the use of single-rank DIMMs, but this information will be discussed separately below, after the explicit mention of working with single-rank modules.

The data on the optimal sizes of one-dimensional arrays for maximizing the bandwidth in STREAM, specified in [**121**], are presented in Table 12. From this table it is clear that with an increase in the number of CCDs and, accordingly, the total capacity of the L3 cache, the optimal size of the arrays increases proportionally. A similar situation occurs with an increase in the capacity of the L3 cache when adding 3D V-cache (in EPYC 9684X).

For 96-core EPYC 9654, when using different numbers of CCDs, different numbers of threads (up to 192), different NPS values, and enabling/disabling boost modes, working with SMT results in a decrease in bandwidth with two exceptions out of all possible combinations. Enabling boost mode resulted in an increase in the achieved throughput with any combination of other parameters, except for cases where 192 threads were used in the benchmark (note that maximum bandwidthes were more often achieved with other number of threads, see below).

Table 13. Memory Bandwidth (MB/s) of 2P Servers with EPYC 9654, EPYC 9684X, or EPYC 9754

Number of CCD	Number of threads	Bandwidth		
		EPYC 9654	EPYC 9684X	EPYC 9754
12	24	739500	717866	
	48	770343	755924	
	96	771153	756739	
	144	774746	755686	
	192	766863	753407	
8	16	651350		755930
	32	771730		782937
	64	784341		787422
	96	779301		780101
	128	765102		774657

Data from [121].

For 128-core EPYC 9754 (Bergamo, CCD=8, NPS=4, thread count up to 128), similar bandwidth behavior with SMT or boost mode enabled also occurs for all parameter combinations, except for benchmarks using 96 or 128 threads.

For 96-core EPYC 9684X with 3D V-cache (CCD=12, NPS=4, thread count up to 192), enabling boost mode always resulted in increased bandwidth, while enabling SMT resulted in decreased bandwidth except when working with 24 threads.

From this data [121] it can be concluded that enabling SMT mode usually reduces bandwidth, while switching the processor to turbo mode (Boost=ON in BIOS) usually increases the achieved bandwidth.

The highest bandwidth values for EPYC 9654 were expectedly obtained at NPS=4. Therefore, we will provide one Table 13 with different CCD numbers for all three processor models with NPS=4, SMT=OFF, Boost=ON. For EPYC 9654, the array size used here was 3.2 GiB, which is optimal for working with CCD=12. Detailed numerical data for other modes (parameters), including the number of active cores, are available in [121].

Bandwidth data with EPYC 9654 using optimal array sizes for each number of CCDs (2.1 GiB for 8 CCDs and 3.2 GiB for 12 CCDs) and dual-rank DIMMs is shown in Table 14.

When using single-rank DIMMs, the achieved bandwidth in this benchmark is noticeably lower. Thus, the “absolute maximum” obtained

TABLE 14. Memory bandwidth (MB/s) in a 2P server with EPYC 9654 at 8 CCDs with 2.1 GiB arrays and 12 CCDs with 3.2 GiB arrays

Number of threads (8/12 CCD)	8 CCD	12 CCD
16/24	656994	742480
32/48	772818	772308
64/96	787566	771964
96/144	784398	774630
128/192	778488	766348

Date from [121].

when working with 8 CCDs and 64 threads with EPYC 9654 is 788 GB/s for a 2P server with dual-rank DIMMs (NPS=4, SMT=OFF, Boost=ON). And when working with single-rank DIMMs with other optimal parameters, the bandwidth is lower, only 726 GB/s.

The data [121] does not show any increase in memory bandwidth achieved in these benchmarks due to the use of 3D V-cache (it is possible that the higher frequency of the EPYC 9654 cores in boost mode has a greater impact on it), and the use of 8 rather than 12 CCDs often yielded higher bandwidth. And the maximum value is achieved with the EPYC 9654 using 64 threads at 96 existing cores in each of the two processors. And if a similar situation exists for other EPYC 9004 models, this may contribute to a decrease in performance scaling with increasing core count across EPYC 9004 models (especially when working with memory-bound applications, such as CFD).

In conclusion of the data analysis [121], it should be noted that although the corresponding benchmarks used arrays of fairly large dimensions, in HPC it may be necessary to work with larger arrays, and it is important to obtain information about whether the bandwidth will decrease in this case.

From other STREAM benchmark data for 96-core EPYC 9004, one can point to [159], where for EPYC 9R14 (roughly speaking, the initial version of EPYC 9654) the dependence of bandwidth on the number of cores involved is demonstrated.

[160] provides data on the change in average bandwidth between all four STREAM benchmarks variants (Copy, Scale, Add and Triad) depending on the number of threads used in a 2P server based on 32-core EPYC 9384X compared to a 2P server with 32-core Xeon Max 9462 using only HBM memory or in combination with DDR in HBM Cache mode. Here, HBM caching was actually considered as an analogue of caching with the larger 3D V-cache in EPYC 9484X. The gain in this bandwith value for the EPYC 9384X compared to the bandwidth of the Xeon Max 9462 in Cache mode decreases as the number of threads increases from 1 to 64. And the gain in this bandwidth in HBM mode without caching increases with the number of threads up to 64, already outpacing the EPYC 9384X at 64 threads.

Other somewhat similar data for the STREAM Triad benchmarks [**161**] are related to Fujitsu PRIMERGY servers (1P with 64-core EPYC 9554P and 2P with 64-core EPYC 9534). There, BIOS settings and selection of NPS=1, 2 or 4 were used, and different variants of supported DIMMs (including RDIMM and 3DS RDIMM) with different values of transfer rates (MT/s) and different capacities were considered. But the main data on relative bandwidth in STREAM are related to NPS=1, disabled SMT and use of 3DS RDIMM (4Rx4 with a capacity of 128 GB).

In this case, a different number of DIMMs per socket is considered. When their number increases to 12, the bandwidth increases, and with 16 modules and more using 2DPC, it decreases by 30-50%. In [**161**], the memory channel interleaving and power consumption during the benchmark execution are also discussed.

Another interesting piece of information is available for 1P and 2P servers (Dell PowerEdge R7615 and R7625) with EPYC 9654P and 9654 processors in the STREAM PTS benchmarks using default BIOS settings [**162**]. Here, the dependence of the achieved bandwidth on the DIMM configurations and, correspondingly, the memory capacity was investigated. It was found that all 4 STREAM benchmarks (Copy, Scale, Add, Triad) showed the same trends, and the data is presented for the Triad benchmark. This data is presented in Figure 14. It shows that the bandwidth increase when transition from 1P to 2P configuration is very close to twofold.

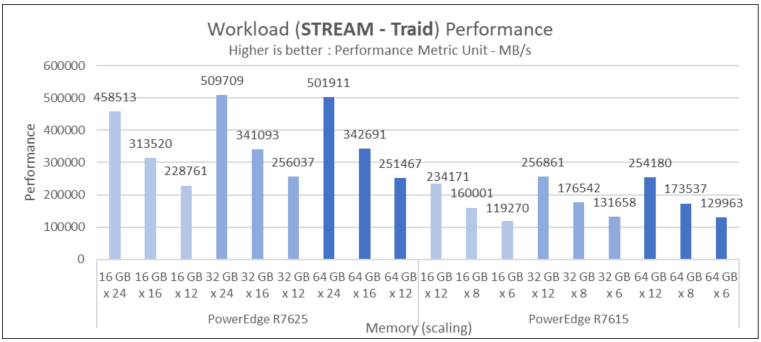


FIGURE 14. Stream Triad bandwidth (MB/s) for servers with EPYC 9654/9654P depending on memory size and DIMM parameters (figure from [**162**])

If we assume a doubling of bandwidth in a 2P server compared to a 1P server based on this data for AMD's reported peak bandwidth of 788 GB/s for the STREAM Triad in a 2P server with an EPYC 9654 (see above), then a very rough estimate of the peak bandwidth of the 1P configuration would be 394 GB/s, which is about 85.5% of the peak value for 12 channels of DDR5-4800 memory (see Table 5).

[162] also provides data similar to Figure 14 for bandwidth per watt and bandwidth per dollar. The best bandwidth and energy efficiency are achieved by a balanced configuration with 12 DIMMs per socket with 32 GB DIMMs. This is the configuration recommended by Dell for memory-bound workloads. This configuration provides 49 percent more bandwidth than 8 DIMMs per socket with 32 GB DIMMs. And a balanced configuration with 12 DIMMs per socket and 16 GB DIMMs provides the best bandwidth per dollar.

Unfortunately, the data presented do not provide important information about the dependence of memory bandwidth on the size of the used arrays. Above, we discussed large memory capacities of arrays (see the data on their optimal sizes in Table 12). As shown in [**163**], the memory bandwidth in a 2P server with a 56-core Xeon Max 9480 reached a maximum at an array size of about 100 MB, then decreased with increasing array size, and stabilized at one GB and higher. But the high-end EPYC 9004 models have a much larger cache capacity than the Xeon Max, and the cache's influence on bandwidth may be stronger.

2.4.3. *Dense Matrix Multiplication (DGEMM) and HPL Benhcmarks*

In terms of approaching peak performance, DGEMM testing is the most important for HPC. According to [**148**], a 2P server with 96-core EPYC 9654 showed about 1.75x better performance in DGEMM (using aocl) than a 2P server with 64-core EPYC 7763 (Milan).

As a base for the EPYC 9004, we will use AMD data from [**121**], where for 2P servers, the DGEMM performance of the EPYC 9654 was 8658 GFLOPS, the EPYC 9684X was 8525 GFLOPS, and the EPYC 9754 was 10730 GFLOPS. This data shows that the slightly higher clock speed capabilities of the EPYC 9654 are more important for performance than the expanded 3D V-cache L3 of the EPYC 9684X, and the use of more cores in the EPYC 9754 is more important than the smaller L3 cache per core in this processor.

AMD's DGEMM performance data for 2P servers with other processor models is presented in Table 15. The data presented in this table are not the maximum values provided by the manufacturer. For example, a later document [**121**] for a 2P server with an EPYC 9654 provides a performance of 8658 GFLOPS (this was already noted above). In addition, there are even higher DGEMM and HPL performance figures for these EPYC 9004 models obtained in benchmarks in China (see, for example, a reference to Chongqing Microcomputer[8]). There, the DGEMM performance of 9283 GFLOPS is provided for the EPYC 9654.

It is also worth noting the DGEMM performance data for EPYC 9004 processors in [**100**], where the DGEMM performance per core was obtained as part of the HPCC benchmarks. The MKL library was used for this. For the core in EPYC 9654, 51 GFLOPS were obtained there, in EPYC 9554 — 58 GFLOPS, and in the 32-core EPYC 9334 — 60 GFLOPS, which is probably due to the increase in the permissible frequency of processors with a smaller number of cores. But the cores performance of mid-range Xeon SPR (in 32-core Xeon 6430) is higher (71 GFLOPS) [**100**].

In addition to the common DGEMM benchmarks, where usually the corresponding module from optimized libraries is used, for the EPYC 9004 there are less interesting data from the Phoronix mt-dgemm benchmarks (see, for example, [**147**]), which use translation of simple nested loops for matrix multiplication without using special libraries, which for such source text must necessarily give lower performance.

[8] https://news.qq.com/rain/a/20230912A0A4M100, accessed 6.03.2025.

TABLE 15. Comparison of 2P server performance in DGEMM and HPL benchmarks with different EPYC 9004 and Xeon ICL processors

SKU	EPYC 9214	EPYC 9174	EPYC 9224	EPYC 9354	EPYC 9374F	EPYC 9534	EPYC 9554	EPYC 9654	Xeon 8380
Number of cores	16×2	16×2	24×2	32×2	32×2	64×2	64×2	96×2	40×2
DGEMM (GFLOPS)	1769	2065	2300	3479	3937	5423	6363	7375	
HPL (GFLOPS)	1742	2017	2262	3411.5	3837	5281	6081	7106	4433

Data from [152], for Xeon 8380 — from [164].

As for the HPL benchmark used in the TOP500, according to [148], the performance of a 2P server with EPYC 9654 is approximately 1.77 times higher than a 2P server with EPYC 7763. In [88], the performance of a similar server with EPYC 9654 is compared with a 2P server with EPYC 7773X, including for different dimensions (from 140 to 220 thousand). For the maximum dimension, the server with EPYC 9654 in HPL is 1.75 times faster. These data were obtained using aocl 4.0 (when using oneAPI MKL 2022.2, the performance was lower).

As with the above-discussed data in Table 15 on the performance of 2P EPYC 9004 servers in DGEMM, the corresponding data for HPL are not the maximum achieved. In a later paper, [121], AMD reported higher values: 8856 GFLOPS for EPYC 9654, 10134 GFLOPS for EPYC 9754, 8620 GFLOPS for EPYC 9684X. The sometimes higher HPL performance of EPYC 9654 and EPYC 9684X than DGEMM is likely due to the presence of an supplementary vector addition unit in the cores in addition to the single FMA unit (see above in Section 2.1).

If we divide the HPL performance shown in Table 15 by the number of cores, then per core for the 40-core Xeon 8380 model we get 55 GFLOPS, for the top 96-core EPYC 9654 — significantly lower, 37 GFLOPS. However, EPYC models with a smaller number of cores can have a significantly higher clock frequency, and for the 64-core EPYC 9554 such performance per core is 48 GFLOPS, and for the 32-core EPYC 9374F it is already higher than for Xeon, 60 GFLOPS (for the low end 16-core EPYC 9214 model — 54 GFLOPS, almost the same as for Xeon). In a certain sense, we can talk

about competitiveness in HPL performance per core between EPYC 9004 and Xeon ICL, so the significantly higher number of cores in the high-end EPYC 9004 models gives them a corresponding performance advantage.

According to AMD [165], the HPL benchmark for a 2P server based on a 128-core EPYC 9754 achieved a performance of 10134 GFLOPS, while for a 2P server with 60-core Xeon 8490H, the performance obtained by Fujitsu was 7296 GFLOPS [166]. The number of cores in the server with EPYC was 2.13 times greater than the number of cores in the server with Xeon, and the performance in HPL was 1.39 times greater. And this performance of the server with 60-core Xeon 8490H is slightly higher than the value given in [88] (7257 GFLOPS) for the server with 96-core EPYC 9654 (the performance presented in [152], indicated in Table 15, is even slightly lower). But the more recent performance values from AMD [121] cited above are still higher.

The review [100] also reported HPL performance data for a 96-core EPYC 9654 (3811 GFLOPS), a 64-core EPYC 9554 (3319 GFLOPS), and a 2P server with mid-range 32-core EPYC 9334 (3527 GFLOPS) using the MKL library, which yielded slightly higher performance than OpenBLAS. The higher performance of the 2P server compared to the 64-core processor is also likely due to boost frequencies. This 2P server also slightly outperformed a 2P server with a mid-range Xeon SPR (Xeon 6430, 3436 GFLOPS) containing the same number of cores, which is likely due to the higher clock rates of the EPYC 9334.

2.4.4. *Performance in HPCG Benchmark*

Another benchmark whose results for supercomputers are included in the TOP500 is HPCG, which is considered by many to be more actual for most HPC applications due to its lower compute intensity and greater sensitivity to memory bandwidth than HPL.

Accordingly, in this benchmark, we can also see the dependence of the result on the memory bandwidth. In [**121**] for 2P servers, the maximum performance, 137.9 GFLOPS, was obtained using a 96-core EPYC 9684X or a 128-core EPYC 9754; with the EPYC 9654, it was slightly lower: 135.0 GFLOPS. These data were obtained when working with dual-rank DIMMs, and when using slower single-rank DIMMs (with processors without 3D V-cache), they was 4% lower. These results were obtained with submeshes of 192 by each axis, with a 60-seconds minimum runtime.

As of 20.11.2024, in the HPCG 3.1 test from PTS[9], 2P servers with EPYC 9654 or EPYC 9754 provided much lower performance under different initial data than in [**121**], while outperforming 2P servers with Xeon ICL (Xeon 8380), but lagging behind the 64-core Xeon SPR (Xeon 8462Y). In this test, data for 2P servers with EPYC 9554, 9654, 9754, and 9684X [**12**] demonstrated a performance advantage over the Nvidia GH200 GPU; the Ampere Altra Max M128-30 lags far behind here. The performance comparison of EPYC 9004 with Xeon SPR and Xeon EMR is further discussed in Section 4.1.

2.4.5. *NAS Parallel Benchmarks (NPB)*

A classic example of memory-bound HPC applications is CFD problems. The NPB benchmark suite is based on CFD-relevant programs, and these programs are also typically memory-bound (see, e.g., [**167**]). An important advantage of NPB is that for each of the NPB benchmarks, which have a two-character name, there is a set of input data for problems of different sizes classes, named with symbols from A (small problem) to F (largest). Classes D, E and F are very large and require 12.8 GB, 250 GB and 5 TB of memory, respectively[10]. Given the support of up to 6 TB of memory in the EPYC 9004, all benchmarks can be run even on a 1P server.

[9] https://openbenchmarking.org/test/pts/hpcg, accessed 11.02.2025.

[10] https://www.nas.nasa.gov/software/npb_problem_sizes.html, accessed 21.11.2024.

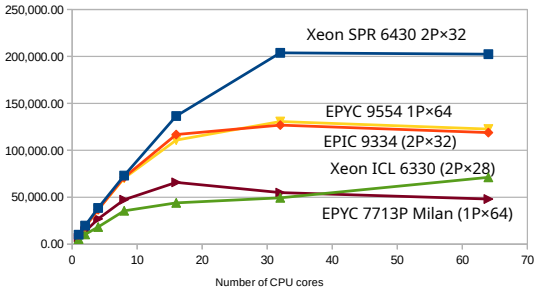


Figure 15. NPB MG.C performance (Mop/s) by the data from [**100**]

The report [**100**] also presented the performance data for the EPYC 9654, 9554, and 9334 in class C tests. The performance of individual cores of these processors in the SP, FT, LU, BT, MG, IS, and CG benchmarks is higher due to higher clock frequencies in the more lower level models with fewer cores than in high-end models (see Table 6), and the cores of all these processors are significantly ahead of the Milan 7713P and Xeon ICL 6330. Similar comparative results are observed in these benchmarks for 1P and 2P servers. However, in the EP test (with extreme parallelism generating a set of N pseudo-random numbers and calculating Gaussian deviations for them), such a strong effect on performance is not observed.

Figure 15 shows the data from [**100**] on the scaling of performance with the number of cores used in the MG benchmark (a multigrid method, perhaps the closest mathematical approximation to CFD problems), and Figure 16 shows the data in the SP benchmark (a scalar pentadiagonal solver). MG.C is memory-bound (see, e.g., [**168**]).

Figure 15 clearly shows that performance gains in the MG C-class benchmark almost stop when using more than 32 cores. And in the SP.C benchmark in Figure 16, we see performance scaling up to 64 cores (and up to 96 for the EPYC 9654). These results may be important for choosing the optimal processor for HPC calculations, as they clearly demonstrate the pointlessness of using processors with a higher core count in some cases.

Since the data for HPCG has already been discussed above, we will not consider the CG benchmark from NPB further here, and the FT benchmark is considered further — in the subsection on FFT. From the results of NPB

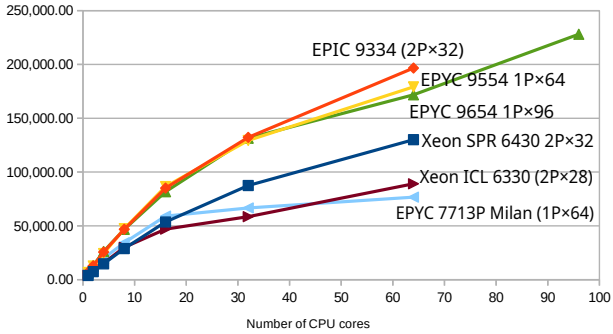


FIGURE 16. NAS SP performance (Mop/s) by the data from [100]

TABLE 16. PTS-benchmarks NPB-1.4 performance

SKU	Number of cores	Benchmark MG.C (Mop/s)		Benchmark LU.C (Mop/s)	
		1P	2P	1P	2P
EPYC 9754	128	127926	247898	281541	598378
EPYC 9654	96	119395	197380	270942	499455
EPYC 9684X	96	136394	233202	342023	537685
EPYC 9554	64	126497	231992	253447	477048
EPYC 9374F	32	122186	214956	179539	372552
EPYC 7763	64	57458	100845	143472	286606

Average values are given for different benchmark results [https://openbenchmarking.org/test/pts/npb Accessed 11.02.2025] as of 24.11.2024

PTSbenchmarks for EPYC 9004 available on the openbenchmarking.org website, we have selected the results of the MG and LU benchmarks for illustration (see Table 16).

These MG.C benchmark results show a large performance increase over the EPYC 7003, performance scaling from 1P to 2P, and the performance gains from the 3D V-cache in the EPYC 9684X. For the EPYC 9554 (1P), the MG.C result in this table is fairly close to what is shown in Figure 15.

On the surface, this table also seems to show performance scaling with the number of cores in the EPYC 9004. But this data also demonstrates a serious shortcoming of the PTS benchmarks results in openbenchmarking.org, which is related to the fact that the data is obtained there only when using all processor cores in each processor.

ItIt is not so important that there are large deviations in the performance of the tests performed by different authors (for MG.C with EPYC 9654 they were more than 10 thousand Mop/s). But the performance of the EPYC 9654 in MG.C (even when for the maximum of all the performance achieved in the tests) turned out to be lower than the 64-core EPYC 9554 (and the 32-core EPYC 9374F with increased frequency cores). The reason for this could also be the cessation of performance growth with the number of cores in the processor over 32, which is shown in Figure 15 — and then the performance could begin to decrease.

Accordingly, the data in Table 16 show the relative performance of different processors in a specific test, without assessing the efficiency of using certain processors. High-end models (primarily focused on HPC EPYC 9654) can show much greater performance relative to models with fewer cores in other tests (or, conversely, not show this).

In addition, it is worth noting the high results of the EPYC 9374F in the MG.C benchmark — they are quite close to the EPYC 9554 with twice as many cores.

The LU.C benchmark also shows a significant performance increase from the EPYC 7003, and performance scaling when transition from processors with lower core counts to processors with higher core counts or from 1P to 2P server configurations. EPYC 9554 servers also outperform Xeon SPR and EMR servers in the MG.C and LU.C benchmarks (see their performance below in Section 4.1).

2.4.6. *Fast Fourier Transform (FFT) Performance Benchmarks*

Next, we consider data for three different FFT benchmarks: FT from the NPB benchmarks, FFTW[11] (more precisely, a variant of FFTW from the FFTE package, part of the HPCC[12] benchmark suite and heFFTe (Highly Efficient FFT for Exascale) [**169**].

[11] https://www.fftw.org/, accessed 23.11.2024.

[12] https://hpcchallenge.org/hpcc/, accessed 8.05.2025.

TABLE 17. Performance in different FFT benchmarks

SKU	heFFTe[1]	NPB FT.C[3]		FFTW from HPCC[4]
	2P	1P	2P	1P
EPYC 9754	208	142074 ± 2376	230807 ± 16627	
EPYC 9654	254	123767 ± 2015	225350 ± 6792	68
EPYC 9684X	323	122140 ± 2543	232995 ± 7382	
EPYC 9554	236	125629 ± 1385	238502 ± 5200	65
EPYC 7773X	125	64292 ± 373	126663 ± 1938	
Xeon Max 9480	2212	—	105720 ± 7286	
Xeon 8490H	129	70543 ± 778	113779 ± 6001	

[1] GFLOPS — for Exascale 2.3 and 2P-configuration EPYC 9004 with a Power mode setting of 400 W. Data from [171].
[2] For work only with HBM.
[3] Mop/s — for NPB 3.4 in the npb.1.4.x variant from [https://openbenchmarking.org/test/pts/npb as of 25.11.2024]. The average values and deviations in different test executions are given.
[4] 4 GFLOPS, data from [100].

The heFFTe library for 3D FFT is originally oriented towards exascaling (and the corresponding American ECP project). Therefore, heFFTe assumes the use of heterogeneous nodes with GPUs, but has also been tested on homogeneous systems, including the Fugaku supercomputer with ARM A64FX [170].

The performance numbers for this benchmark[13] depend on the Exascale variant they are presented for (the latest version as of 2024-11-24 is 2.4), on the heFFTe version (the latest version as of 2024-11-24 is 1.1.0). They use the r2c benchmark with different backends (we are looking at FFTW), dimensions N on all axes (Table 17 shows for N=512) and data formats (Table 17 shows heFFTe data for FP64). The link above provides data for different generations of Xeon and EPYC.

The heFFTe benchmark here shows a big step up from Zen 3 to Zen 4, a certain scaling of performance to 96 cores in a processor (only in the very limited sense indicated above in the subsection on NPB benchmarks). And the 128-core EPYC 9754 with a reduced L3 cache showed lower performance than the 96-core EPYC Genoa), which is combined with both the large positive effect of the 3D V-cache in the EPYC 9684X and the high performance increase when working with high-speed HBM memory in the Xeon Max. The latter corresponds to the FFT performance being limited by memory bandwidth.

[13] https://openbenchmarking.org/test/pts/heffte, accessed 24.11.2024.

The NPB FT.C benchmark data provided is less informative, since the data from different authors gave large deviations. But here you can see a large increase in the performance of the EPYC 9004 compared to the 64-core EPYC 7773X (Milan), scaling of performance when upgrading from 1P to 2P server configurations, and, to a very limited extent, scaling in the EPYC 9004 with an increase in the number of cores in processors from 64 in the EPYC 9554 to 128 in the EPYC 9754 (despite the reduction in the L3 cache capacity in it). In addition, there is a less significant positive effect here from both the L3 cache capacity and the high-speed HBM memory (in the Xeon Max).

2.4.7. *Performance in Continuum Mechanics*

This subsection will mainly deal with CFD problems, and aerodynamics is also included here due to the proximity of the mathematical methods used, but the available performance data in it mainly concerns weather prediction only.

The performance data discussed here is comparable across the different processor models discussed in this review; therefore, it mainly concerns performance in known applications. As an example of research in this area that is not discussed here, we will mark the articles articles where using servers with EPYC 9654 [**172**, **173**].

However, we also note here the work [**174**], which is in some sense also related to computational chemistry (it concerns chemical combustion problems), where plugins from CFD are used (including from the Nek5000 application), and the performance on a 2P server with EPYC 9654 is studied in comparison with the use of Nvidia H100 and AMD MI250X GPUs.

Studying performance in the CFD area is of interest for modern multi-core processors, including for understanding the efficiency of using processors with different numbers of cores, since memory-bound CFD applications may stop scaling performance at a number of cores far from the maximum currently available (see, for example, the NPB MG.C benchmark data above).

CFD performance is one of the most important metrics for the EPYC 9004 in HPC, as CFD is probably where the previous two EPYC Zen generations were used more often than in other HPC areas. A demonstration of the performance advantages of the EPYC 9654, 9554, and 9374F over

the Zen 3 models (EPYC 7763 with 64 cores and EPYC 75F3 with 32 cores) can be found, for example, in [175]. There, even the 32-core EPYC 9374F, which has the lowest performance among the EPYC 9004 processors used, outperformed both Zen 3 models.

Immediately after the start of deliveries of servers with Zen 4 processors, a general recommendation appeared on a well-known CFD website focused on the practical application of fluid dynamics problems about choosing EPYC 9004 models for their use in this area [155].

Since the cost of the license for the used CFD software is often proportional to the number of processor cores available for use, the cost/performance ratio takes on a higher practical significance. It is clear that it is recommended to use processor models that achieve a higher memory bandwidth, respectively supporting 12 DDR5 channels. In addition, the L3 cache capacity is also noted as important, respectively the recommendation is limited to the Genoa series (including Genoa-X using 3D V-cache). It is not the top 96-core EPYC 9654 model that is recommended there, which is primarily due, perhaps, to the higher cost indicator.

The data in Figures 15 and 16 above also correspond to the choice of processors with not the largest number of cores. This choice is partly consistent with the previously presented data for the STREAM Triad test, where the highest memory bandwidth in a 2P server with EPYC 9654 was obtained using 96, not 192 threads when using 12 CCDs, and when working with 8 CCDs (and 64 threads), the bandwidth was even slightly higher.

Performance of OpenFOAM tools. We begin the application-specific part of the CFD performance discussion with data for more general software, since OpenFOAM is used in general for solving partial differential equations, and as noted above in Section 1, is applicable to more than just continuum mechanics problems. The latest version of OpenFOAM as of 2024 was 12 [176].

Various benchmarks have long been available[14] for OpenFOAM, and numerous data on its scaling depending on the number of processor cores for a wide variety of processors, both x86 and ARM, constantly appear in the OpenFOAM-oriented blog of CFD users — from 2018 to the present.

[14] https://www.cfd-online.com/Forums/hardware/198378-openfoam-benchmarks-various-hardware.html, accessed 1.12.2024.

The currently used HPC-oriented benchmarks were developed in 2019 [177]. They are used for different purposes and have different grid sizes. A full list of them is available in [178]. The data from these benchmarks[15] began to appear actively for modern processors when they were included in the PTS, where the openfoam-1.2.x benchmark results for the latest processors refer to OpenFOAM 10. For these test variants, we will consider the data from [171] below, since this is the best fit for comparing different processor models.

In [179], AMD provides average composite values for different OpenFOAM benchmarks only for the relative performance of 2P servers with EPYC 9004 compared to Xeon SPR. Compared to a 2P server with 56-core Xeon 8480+, a server with EPYC 9654 is 1.55 times faster (and has 1.7 times more cores), and with EPYC 9684X, the server is 2.08 times faster. And for the 60-core Xeon 8490H, the corresponding ratio is 1.02. In addition, compared to a 2P server with 32-core Xeon 8462Y+, a server with also 32-core EPYC 9384X is 1.77 times faster. It should also be noted from these data that there is a significant increase in performance due to the use of 3D V-cache, which is in line with the recommendations [155].

What may be other important in the data [179] is that AMD showed superlinear performance scaling for a cluster with Infiniband HDR of servers with EPYC 9684X at up to 8 nodes. This is not surprising, since this was already the case with EPYC processors of the Zen 2 (Roma) architecture, which also use a microarchitecture hierarchy using CCX, and superlinear scaling was shown in OpenFOAM [180, 181]. Superscalar scaling there is explained by more efficient use of the large L3 cache capacity in the cluster nodes. At the same time, according to [180], higher performance is achieved by using only half of the cores in 2P nodes with 64-core EPYC 7742, which contributes to high-performance work with the L3 cache.

It is also useful to note here that superlinear speedup in clusters is also achieved by sparse matrix-vector multiplication (SpMV) [182], and this operation is actively used in CFD tasks.

Good scaling in the OpenFOAM MotorBike benchmark with different mesh sizes and up to 16 nodes is shown also in a cluster with Infiniband NDR and 32-core EPYC 9354 in 2P nodes [183].

And as numerical performance indicators of 2P servers with different high-end EPYC 9004 models for a concrete OpenFOAM benchmark, we present in Table 18 the data [171] for the drivaerFastback benchmark using a medium-sized grid.

[15] https://openbenchmarking.org/test/pts/openfoam, accessed 30.11.2024.

TABLE 18. Performance in the OpenFOAM drivaerFastback benchmark

SKU	Number of cores	Runtime (seconds)
EPYC 9684X	96×2	84
EPYC 9654	96×2	106
EPYC 9754	128×2	115
EPYC 9554	64×2	127
EPYC 7773X	64×2	165
Xeon 8490H	60×2	192

Data for servers with EPYC 9004 is obtained in POWER mode at 400W.

The data in this table shows that the high-end EPYC 9004 models are significantly ahead of the top Zen 3 model, which in turn significantly outperforms the top Xeon SPR model. In addition, the positive impact of 3D V-cache on performance in the EPYC 9684X is visible. Data comparing OpenFOAM performance on the EPYC 9004 and Xeon Max is provided below in Section 4.

Performance in CFD applications. In relation to the CFD applications from ANSYS, which are widely used in the world, when used in servers with EPYC 9004, including in clusters, there are general proposals from Supermicro [**184**]. They recommend disabling SMT and using NPS=4. As for the inexpediency of using SMT in different processors for CFD tasks, this has been known for quite a long time (see, for example, [**185**]).

For four ANSYS applications — LS-DYNA, CFX, Mechanical and Fluent in [**186**] presents data on the relative performance of 2P servers with 64-core EPYC 9554 or 32-core EPYC 9374F compared to a 2P server based on a 32-core EPYC 75F3. For LS-DYNA, the acceleration is up to 1.8 or 1.4 times, respectively, for CFX — up to 1.9 or 1.6 times, for Mechanical — up to 1.9 or 1.5 times, for Fluent — up to two or one and a half times.

For the LS-DYNA, CFX, and Fluent applications in different benchmarks, quantitative rather than relative performance indicators are presented in [**187**] for 2P servers with 32-core EPYC 9384X and 96-core EPYC 9684X. The choice of models with 3D V-cache is related to the facts of performance increase when using it. And the use of 32- and 96-core models in the benchmark allows us to evaluate the performance for both high-end and mid-range processors.

A wide range of benchmarks were used for performance evaluations in [187] — 3-Cars, Neon, ODB 10m for LS-DYNA; LeMans car, Automotive Pump and Airfoil 10m, 50m and 100m (numbers indicate how many millions of cells are applied) for CFX; 15 different benchmarks for Fluent.

In [187], the performance of these models was compared with the Xeon Max 9462 (32 cores) and the top 56-core Max 9480 model, which contain 64 GB of high-speed HBM memory per processor. For almost all of the benchmarks used, this capacity was sufficient for 2P servers. The exceptions were AirFoil 100 for CFX and 3 benchmarks for Fluent. The performance increase when switching to work only with HBM from the Cache mode in Xeon did not exceed 5%. And in Cache mode, the average composite performance acceleration of EPYC relative to Xeon for these three applications for 32-core processors varied from 1.3 to 1.7, for high-end models — from 1.8 to 2.3 times.

There is many data presented by AMD on the performance acceleration in 2P servers with EPYC 9004X relative to Xeon SPR. For 32-core EPYC 9384X relative to Xeon 8462Y+: in CFX with Airfoil 10m — up to 2 times, in Fluent with Pump 2m — up to 1.8 times, in LS-DYNA with 3-cars — up to 1.9 times [165]. Accelerations in 2P servers with EPYC 9684X compared to 2P servers with Xeon 8480+ (top models), as well as in 2P servers with 32-core EPYC 9384X processors compared to Xeon 8462Y+ for LS-DYNA are presented in [188].

For CFX, similar comparative data for 2P servers with EPYC 9684X and 9654 relative to 64-core EPYC 7773X, 56-core Xeon 8480+ and top 60-core Xeon 8490H, as well as for 32-core EPYC 9374F relative to Xeon 8462Y+ are given in [189]. And for the Fluent application in [190], similar data are given for the relative performance of 2P servers with 96-core EPYC 9684X, 9654, 32-core EPYC 9384X, 9374F, EPYC 7773X processors and with Xeon 8480+, 8490H, 8462Y+. It also demonstrates super-linear performance scaling across a range of Fluent benchmarks in a cluster with Infiniband HDR and 2P servers with EPYC 9684X at up to 8 nodes.

In addition to the ANSYS applications widely used in the world discussed above, it is also impossible not to mention the performance data of various EPYC 9004 processors in the famous software package of the Lawrence Livermore National Laboratory (USA), LULESH[16], which is aimed at exascaling. Although this application is narrowly focused

[16] https://asc.llnl.gov/codes/proxy-apps/lulesh, accessed 13.12.2024.

on solving a specific Sedov explosion problem, it uses numerical algorithms that are quite typical for other scientific applications for CFD. In LULESH, to solve the equations of hydrodynamics, the entire spatial domain is partitioned into a collection of volumetric elements defined by a mesh.

The corresponding results for 1P and 2P servers with EPYC 9004[17], also in [**147**], show poor performance scaling when replacing one processor model to another with a higher number of cores, for example, from EPYC 9554 to EPYC 9654, which requires first considering performance scaling with a different number of threads per processor. The only thing is that the data available there shows higher performance of servers with these processors compared to EPYC Zen 3 and with Xeon ICL, Xeon SPR, Xeon Max and Xeon EMR.

Another interesting example is the performance in the CFD area for the MFC application, which is designed to solve compressible multiphase flow problems — i.e., not related to such widely used applications as the ANSYS applications discussed above. In [**191**], data is presented on the performance achieved in MFC with Nvidia GH200, H100, A100, and V100 GPUs, as well as AMD MI250X — including in relation to the performance of 1P servers with different processors, including EPYC 9654, Xeon Max 9468, and Nvidia Grace (ARM Neoverse v2). Among the processors, the EPYC 9654 turned out to be the fastest, lagging behind the GPU by 1.5-5.3 times.

In conclusion, this subsection presents data on the performance of the WRF application, intended for both atmospheric research and operational weather forecasting.

WRF is a memory-bound application. All WRF benchmark data[18] presented below are for the Conus 2.5km dataset from one of the laboratories of the National Center for Atmospheric Research in the USA. In particular, they are used in the WRF benchmark from PTS[19], where WRF 4.2.2 was used. Such benchmarks were conducted, for example, as part of the PTS benchmarks in [**147**, **171**].

According to AMD [**192**], in 2P servers, the performance of EPYC 9684X and EPYC 9654 is more than twice as high as EPYC 7773X, and 1.7-1.8 times higher than Xeon 8480+. At the same time, EPYC 9684X with 3D V-cache has the highest performance.

[17] https://openbenchmarking.org/test/pts/lulesh, accessed 13.12.2024.

[18] https://www2.mmm.ucar.edu/wrf/users/benchmark/v44/v4.4_bench_conus2.5km.tar.gz, accessed 4.12.2024.

[19] https://openbenchmarking.org/test/pts/wrf, accessed 4.12.2024.

Table 19. WRF 4.2.2 benchmark data (with conus 2.5km) for servers with EPYC 9004

	EPYC 9654		EPYC 9554		EPYC 7773X	
	1P	2P	1P	2P	1P	2P
Runtime, s	7370	3940	8228.5	4503	15436	7411

The numbers in Table 19 are obtained in [**147**] using Power determinism for EPYC 9004. They demonstrate a high performance increase already when replacing 64-core EPYC 7773X processors with EPYC 9554. In [**171**] higher performance of a 2P server with EPYC 9684X (but also in higher power mode) is shown, which also outperforms Xeon 8490H.

Performance data for WRF 4.2.2 on 1P and 2P servers with 32-core EPYC 9374F processors in the same benchmark, including comparisons with the performance of servers with 32-core EPYC Zen 3 and Xeon ICL, is available in [**193**].

According to [**192**], the performance of a cluster with Infiniband HDR and 2P servers based on EPYC 9684X and EPYC 9654 up to 8 nodes grows almost linearly (with EPYC 9684X — slightly faster, with EPYC 9654 — slightly slower). In [**183**], the performance of WRF 4.5 in an Infiniband NDR cluster with 2P servers using 32-core EPYC 9354 showed almost linear scalability with the number of nodes in the cluster up to 16.

2.4.8. *Performance in Computational Chemistry Problems*

This section covers areas that can be applied to all areas of chemistry, including biochemistry and materials science. Performance data for molecular dynamics, quantum chemistry, and molecular docking are covered here. However, since computational chemistry is also being actively used for drug design, we also add information at the end of this section about genomics, which has become actual for medicine, using big data and possibly vector operations.

Molecular dynamics. Molecular dynamics problems have become one of the most important areas for which CPU and GPU developers demonstrate performance data. One of the most important reasons for this is the high level of parallelism that can be achieved. But in terms of the workloads used, there are also fundamental differences in this area depending on whether classical or quantum molecular dynamics is used, and in the latter, also depending on the basis used (plane waves or, less commonly, Gaussian functions). Most often, benchmarks are performed for

classical molecular dynamics, which will be discussed further. Data for quantum molecular dynamics are discussed at the end of this subsection, with an explicit iniication of the quantum level of calculations.

AMD provided special performance reviews for its EPYC 9004 processors for molecular dynamics tasks. In [**194**], acceleration is demonstrated in the well-known by high parallelization level GROMACS application on an Amazon AWS EC2 hpc7a.96xlarge computing instance with EPYC 9004 compared to EPYC 7003 in hpc6a.48xlarge and a good level of performance scaling when using up to 16 virtual machine instances. And in [**195**], AMD showed acceleration on a 2P server with a 128-core EPYC 9754 (Bergamo) compared to a similar server based on a 56-core Xeon 8480+ by more than two times on classical molecular dynamics tasks (in the GROMACS and NAMD software suites), which is only slightly lower than the increase in the number of cores. For quantum molecular dynamics (with the DFT method in the plane wave basis) in Quantum Espresso, the speedup was 1.4 times. It is worth noting that AMD demonstrated this using Bergamo with a reduced L3 cache capacity per core, although it offers such processors primarily for cloud technologies.

In [**121**], AMD provided data on the performance of a 2P server with EPYC 9654 in classical molecular dynamics using the GROMACS program for calculating a molecular system (with 1536 thousand atoms) of water molecules, showing the dependence on the inclusion of boost frequencies, SMT, the use of different NPS and different numbers of cores in the CCX, forming a common L3 cache there (see Figure 7). These data are presented in Figure 17.

The ordinate axis here shows the achieved performance — the number of calculated nanoseconds (ns) in the time trajectory per day. And on the horizontal axis, for each NPS value and each number of used CCDs (8 or 12), additionally number of cores involved in the CCX (from 1 to 8) is indicated. These data show that maximum performance can be achieved in boost mode with SMT disabled. It is shown here that using finer NUMA settings (higher NPS levels) is not necessary. The dependence on the number of used cores helps the user to determine a more optimal processor in terms of their number.

In [**183**], the scalability of GROMACS in a 16-node cluster with Infiniband NDR 200 and 2P servers with 32-core EPYC 9354 with different inputs (using 5 different molecular systems) was obtained. The corresponding results are presented in Figure 18, which shows good scaling in all cases. The ordinate axis here shows the performance, as in Figure 17.

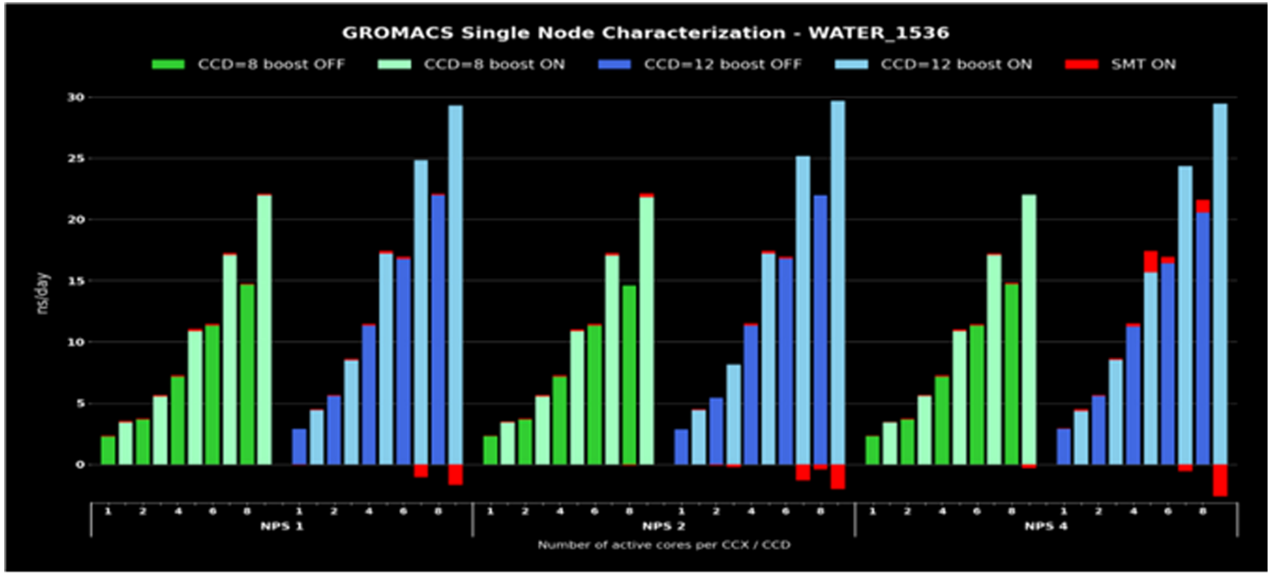
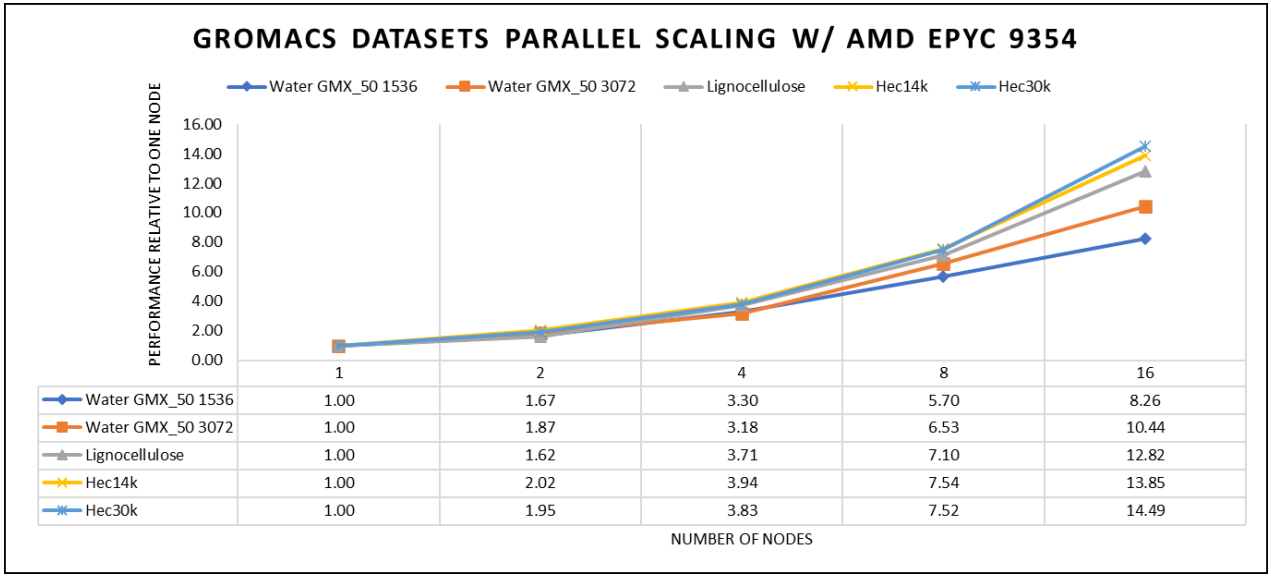


Figure 17. GROMACS performance in 2P server with EPYC 9654: dependence on server parameters. Figure from [121]



	1	2	4	8	16
Water GMX_50 1536	1.00	1.67	3.30	5.70	8.26
Water GMX_50 3072	1.00	1.87	3.18	6.53	10.44
Lignocellulose	1.00	1.62	3.71	7.10	12.82
Hec14k	1.00	2.02	3.94	7.54	13.85
Hec30k	1.00	1.95	3.83	7.52	14.49

FIGURE 18. GROMACS performance scaling in a cluster with EPYC 9354; Figure from [183]

Table 20. AMBER parallelization on a 2P server with EPYC 9654

Number of parallel processes	Performance (ns/day)	
	Cellulose NVE	Nucleosome GB
16	3.31	0.46
32	6.10	0.93
64	10.01	1.85
96	11.99	2.55
128	12.26	3.26
160	11.49	3.75
192	10.58	4.00

The speedup achieved during parallelization depends, in particular, on the size of the system, which is clearly visible when comparing parallelization in benchmarks of water molecule complexes: Water_GMX50_1536 (1536k atoms) and Water_GMX50_3072 (3072k atoms): the latter, containing over 3 million atoms, is parallelized significantly better.

The performance of 1P and 2P servers with EPYC 9684X, 9654 and 9554 in GROMACS with such water molecule complexes is presented in PTS benchmarks[20]. The data available there for the Water_GMX50_bare tests in GROMACS 2024 demonstrate the higher performance of the high-end EPYC 9004 models compared to Xeon SPR, Xeon Max and Xeon EMR.

Here it seems relevant to provide data on the dependence of the EPYC 9654 performance in molecular dynamics on the number of parallel processes involved in the 2P server, which are available in [88] for the AMBER software package for calculations in the Cellulose NVE benchmarks with explicit solvent accounting (there the molecular system contains about 400 thousand atoms) and Nucleosome GB with an implicit solvent (there about 25 thousand atoms). The corresponding data from [88] are presented in Table 20.

They show that in both benchmarks good performance scaling stopped when using more than 64 parallel processes, which is why it was recommended to choose the corresponding number of cores. In addition, the data in this table show that the achievable level of parallelism here depends not only on the size of the molecular system.

[20] https://openbenchmarking.org/test/pts/gromacs, accessed 10.12.2024.

TABLE 21. Performance of servers with EPYC 9004 in NAMD 3.0b6 in the STMV benchmark

SKU	Cores	Frequency, GHz	Price	Average performance, ns/day	
				1P	2P
EPYC 9124	16	3–3.7	$1083		1.6
EPYC 9184X	16	3.55–4.2	$4928		1.8
EPYC 9224	24	2.5–3.7	$1825		2.2
EPYC 9374F	32	3.85–4.3	$4850		3.3
EPYC 9534	64	2.45–3.7	$8803		3.9
EPYC 9634	84	2.25–3.7	$10304		5.0
EPYC 9654	96	2.4–3.7	$11805		5.6
EPYC 9684X	96	2.55–3.7	$14756	3.3	6.4
EPYC 9734	112	2.2–3	$9600		5.6
EPYC 9754	128	2.25–3.1	$11900	3.0	6.0

Prices — from [49] as of 11.02.2025

Report [88] also provides data showing higher performance of the AMBER software suite on this server compared to a 2P server with 64-core EPYC 7773X and a 2P server with 24-core Xeon ICL with the same number of parallel processes. However, the Xeon ICL was faster than both EPYC servers in the Nucleosome GB benchmark with up to 32 processes.

Performance data for 1P and 2P servers with 96-core EPYC 9654 and 9684X in another well-known application, LAMMPS, is available in PTS benchmarks[21]. And performance scalability data for LAMMPS in a 16-node cluster with Infiniband NDR 200 and 2P servers with 32-core EPYC 9354 with different initial data was obtained in [183].

The last thing we will discuss in terms of classical molecular dynamics performance data is the NAMD application. From the NAMD PTS benchmark results[22], we selected the STMV data, which pertain to a molecular system containing over 1 million atoms. These results are presented in Table 21, which shows the average performance among the calculations performed. Given the deviations from the average in these results, the performance values presented have been rounded.

From the data in the table, we can say that the achieved performance increases with the number of cores (although in a concrete model, the performance curve on the number of cores used may bend down from linear

[21] https://openbenchmarking.org/test/pts/lammps, accessed 11.12.2024.
[22] https://openbenchmarking.org/test/pts/namd, accessed 4.12.2024.

as their increases). Increasing the L3 cache capacity due to the use of 3D V-cache causes an increase in performance, and this effect stronger than the effect of increasing the number of cores above 96. It is clear that it would be better to look at the performance dependence on the number of cores used in the calculation for each concrete model, since this could also demonstrate a decrease in performance when their number increases above a certain threshold — as was the case, for example, at 64 cores in AMBER.

Of course, one cannot draw any clear conclusions from this data. For example, using Bergamo EPYC 9734 instead of Genoa EPYC 9654 even exclusively for NAMD calculations does not make sense, if only because the results may depend on the calculated system, and the deviations in different calculations from the average performance in a 2P server with EPYC 9654 were abnormally high compared to other models — about 0.4 ns/day.

Additionally, various relative performance data is available in NAMD benchmarks comparing different EPYC 9004 models to Xeon SPR, Xeon EMR, and Xeon Max (see, for example, data from AMD [**160**, **195**, **196**]), showing the performance advantages of these EPYC models relative to the indicated series of Xeon processor.

Performance data for different EPYC 9004 models in quantum molecular dynamics is available for CP2K and Quantum Espresso applications. Performance data for EPYC 9004 servers in various CP2K benchmarks using DFT in the plane-wave basis (including the linear scaling variant) in the quantum chemical part of calculations is also available[23]. And data on the relative performance of Quantum Espresso in a 2P server with EPYC 9754 and EPYC 9654 compared to a 2P server with Xeon SPR 8480+ AMD provided in [**195**, **196**].

Quantum chemistry. Now we can move on to the performance in quantum chemistry problems. In [**88**], the performance in calculations for the VASP application was investigated using the DFT method in the plane wave basis using different pseudopotentials and exchange-correlation functionals for a molecular system of a thousand atoms. The calculations were performed on 2P servers using 96-core EPYC 9654, 64-core EPYC 7773X (Milan) and 24-core Xeon ICL. The Intel compiler and MKL library were used to generate the executable code, without AVX-512 support.

[23] https://openbenchmarking.org/test/pts/cp2k, accessed 12.12.2024.

When compared with the same number of parallel processes in different servers and one process per core, using up to 32 cores (on two processors), Xeon ICL was faster (it was ahead of Milan even when parallelizing using all 48 cores). EPYC 9654, naturally, was ahead of Milan with the same number of cores involved in parallelization.

But such calculations require a high memory bandwidth, for example, due to the use of FFT. The calculation time with the EPYC 9654 was usually reduced when using only up to 64 parallel processes (in one calculation — up to 128), and then it outperformed the Xeon ICL.

In [88], the performance of another widely used quantum chemical software suite, Gaussian 16, was also investigated in these servers. There, the calculation was performed for the organic molecule valinomycin (168 atoms) with the widely used DFT method (with test0397, part of supplied with Gaussian 16) in the 3-21G Gaussian basis set. With 48 parallel processes, the calculation time on the server with EPYC 9654 was 45 seconds, which is 1.2 times faster than with EPYC 7773X, and 1.6 times faster than with Xeon ICL. But it should be said that such calculations clearly do not require the use of modern powerful processors.

When using a more accurate and more widespread 6-31G (d,p) basis set in such a calculation, the execution time using 48 parallel threads increased to 164.5 seconds, and decreased with an increase in the number of threads used — to 89 seconds on 192 threads.

The performance of Gaussian 16 in the last benchmark scaled decently with the number of threads up to 96. Considering the not very high rates of parallelization achieved in this software suite and the computational intensity of quantum chemical calculations, it can be assumed that such calculations in the traditional Gaussian basis of larger molecular systems and/or by more accurate methods in servers with multi-core EPYC 9004 will be effective, which corresponds to the conclusion in [88]. However, it would be desirable to justify this with corresponding concrete calculations.

Other data on the execution times on EPYC 9654 in the TDDFT calculation using a numerical basis centered on atomic nuclei (these are computationally more complex calculations than in DFT with a plane wave basis set), using the FHI-AIMS software package for calculating the electrical conductivity of warm dense hydrogen are available in [197].

For the “built-in benchmark” of CP2K application (H2O-DFT-LS), a linearly scalable DFT calculation, AMD reported in [160] the speedups

achieved in 2P servers with EPYC 9004 processors containing 3D V-cache compared to 2P servers with Xeon Max (with HBM memory). With 32-core processors, the EPYC speedup was reported to be about 1.8x (for EPYC 9384X compared to Xeon Max 9462); for the top 96-core EPYC 9684X compared to the 56-core Xeon Max 9480, it was about 2.1x.

And in [**183**], the scaling of the performance of the quantum chemical application CP2K in a cluster with 2P servers with 32-processor EPYC 9354 and Infiniband NDR 200 is demonstrated. When calculating systems of water molecules within the "built-in" H2O-DFT-LS-NREP and H2O-64-RI-MP2 (using the RI-MP2 method) benchmarks, a good acceleration of performance was achieved with an increase in the number of nodes to 16 (superlinear in the H2O-64-RI-MP2 benchmark).

In conclusion of the analysis of the performance data of quantum chemical applications on EPYC 9004, we present information for the NWChem application, which is usually distinguished by the more highest achievable level of parallelism. The corresponding information became available[24] due to the inclusion in the PTS of the famous NWChem benchmark using the DFT method in the 6-31(d) basis for the C240 molecule (3600 basis functions), without taking into account symmetry and without geometry optimization. This benchmark is distinguished by good parallelism in clusters and, in particular, on the Cascade supercomputer (ranked 13th in the November 2013 TOP500 list) [**198**]. When using two 16-core Intel Xeon E5-2670 processors per node (Intel Xeon Phi, judging by this publication, was not used here) with 128 cores, 4 iterations of the SCF took about 500 seconds, and performance scaling was observed up to the use of 2048 cores.

In the NWChem PTS benchmarks[25], [**199**, **200**] the same method for the same molecular system showed poor performance scaling with increasing number of cores used, for example, when "moving" from a 1P to a 2P server with high-end EPYC 9004 models (and when replacing a 32-core EPYC 9374F processor, for example, to EPYC 9554 or 9654). The reasons for this require further study.

Molecular docking. For EPYC 9004, there is data on a significant increase in miniBUDE application performance when enabling AVX-512 support [**150**], and in [**201**] much higher performance in miniBUDE is

[24] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.
[25] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.

demonstrated for a server with EPYC 9654 compared to servers with Xeon ICL and Xeon SPR 8490H (relative to the latter — more than twice). Xeon EMR 8592+ also lags far behind EPYC 9654 in performance [202]. Data on miniBUDE performance in servers with different models of Zen 4 and Zen 3 processors is available[26].

Genomics. Of course, genomics tasks are not related to computational chemistry tasks, but rather to molecular biology. When using computers, we are actually talking about bioinformatics or computational biology. But genomics is highlighted in a separate subsection here because the huge area of computational chemistry is important for drug design tasks, and modern genomics is also used in medicine. Servers and clusters with high-end EPYC 9004 models have been actively used in scientific research in genomics (see, for example, [203–205]. For a classic application in this area, GATK [206], AMD indicates a 28% increase in performance in a 2P server with EPYC 9654 compared to a 2P server with Xeon 8490H [196].

The later Sentieon application has shown better performance in genome sequencing [207], and [208] showed that on a 2P server with an EPYC 9654, the overall execution time was faster than when sequencing with the Parabricks application [209] on a server with an Nvidia H100 GPU.

2.4.9. *Benchmarks in AI Tasks*

Despite the fact that AMD, unlike Intel, may not be as active in the field of AI, both the EPYC 9004 hardware has the necessary resources (including due to support for AVX-512), and the company offers its own developments in the software field (the ZenDNN library as an analogue of oneDNN). Naturally, frameworks are also available for the EPYC 9004, including TensorFlow and PyTorch, which are united by AMD as part of the UIF (Unified Inference Frontend) tools [210]. And for tuning the EPYC 9004 for AI tasks, AMD has prepared a corresponding guide [140]. And almost immediately after the appearance of these processors, scientific publications in the field of AI began to appear using servers based on the EPYC 9004 with an analysis of data on the achieved performance (see, for example, [211, 212]).

According to [150], when enabling AVX-512 support in EPYC 9654, the performance in the ResNet-50 model using TensorFlow 2.10 when working in the BF16 format increases by 73%.

[26] https://openbenchmarking.org/test/pts/minibude, accessed 15.12.2025.

AMD reports a 68% performance (images per second) improvement on the well-known Yolo (You look only once) v5 [213] object detection model using PyTorch, and a 97% improvement (for image recognition latency) with the ResNet-50 model and BF16 format when using also ZenDNN [210]. This performance improvement was achieved on a 2P server with an EPYC 9654.

[214] provides AMD data on the performance of a 2P server with EPYC 9654, including relative to a 2P server with 64-core EPYC 7763, using ZenDNN for three different AI models. ResNet-50, running in FP32 format on data from the ImageNet image database using the WordNet hierarchy (resnet50_fp32_pretrained_model.pb), achieved inference performance of about 920 images/sec with a batch size of 640, or 927 images/sec with a batch size of 960. This is about 2.1x faster than the Zen 3 server.

For natural language processing, the pre-trained BERT-Large model with 340 million parameters for wwm_uncased_L-24_H-1024_A-16 in FP32 format achieved a performance of about 29 samples per second for sequence length 256, or about 19 samples per second for sequence length 384. This is about 1.8x faster than on a Zen 3 server. For the deep learning recommendation model, using the DLRM benchmark from the well-known MLPerf Inference [215] benchmark set for AI (for tb00_40M.pt in FP32 format), performance scores[27] were also obtained that were not submitted for official verification to MLCommons for posting on the website [214]. Performance with the EPYC 9654 was around 2948 samples per second for batch size 1 and around 3132 samples per second for batch size 2, which is 1.7-1.8x faster than on the server with Zen 3 [214].

Another well-known benchmark is for OpenVINO[28], an open-source, cross-platform deep learning toolkit (aimed at high-performance inference) being developed by Intel [216], which already supports a range of well-known models across a range of categories, including Large Language Models (LLM), computer vision, and generative AI.

Table 22 shows the performance data for the AI inference tasks in the OpenVINO 2022.3 PTS benchmarks, which measure the number of inferences (frames) per second, FPS. Three different models were used: Vehicle Detection (VD in the table), Person Vehicle Bike Detection (PVBD in the table), and English-to-German machine translation (ENtoDE in the table). All benchmarks used 2P servers and the FP16 format.

[27] https://mlcommons.org/benchmarks/, accessed 8.05.2024.

[28] https://www.intel.com/content/www/us/en/developer/tools/openvino-toolkit/overview.html, accessed 28.11.2024.

TABLE 22. Performance (in FPS) data in OpenVINO benchmarks

SKU	Number of cores	Model VD	Model PVBD	Model ENtoDE
EPYC 9754	128×2	7097	10290	1249
EPYC 9654	96×2	7970	10005	1001
EPYC 9684X	96×2	8125	9766	1065
EPYC 9554	64×2	6270	7536	805
Xeon Max 9480	56×2	3607	7652	688
Xeon 8490H	60×2	3679	7617	693

Data from [171], for EPYC obtained in POWER mode with 400 W, for Xeon Max — using only HBM memory.

According to better known data for different GPUs, AI tasks are characterized by a high level of parallelization and are memory bound. Accordingly, thread competition for memory access can lead to performance degradation and a decrease in the level of parallelization. The latter may occur not only in the VD model when replacing EPYC 9654 to EPYC 9754, but this requires further study. Table 22 does not provide data [171] on latencies, since this indicator is not very acceptable in terms of its dependence on the number of cores involved in the calculation.

The use of 3D V-cache does not always provide a performance advantage. The often strong performance advantage of EPYC 9004 models compared to Xeon SPR and Xeon Max may be due to the software tools (including frameworks) used there in Xeon servers.

The possible conclusions from this table are extremely limited. Naturally, its data only indicate that the EPYC 9004 can often outperform the Xeon SPR in these benchmarks. There are, of course, the results of the OpenVINO PTS benchmarks, where the Xeon 8490H outperformed the EPYC 9004 (see, for example, [217]).

Other performance data using OpenVINO tools was obtained by AMD for (developed by Microsoft) DeepSpeed library [218] using several models for LLM, including opt-350m (350 million parameters) and opt-1.3b (1300 million parameters) with batch sizes ranging from 1 to 128 [219]. This data is presented in Figure 19, where the performance (in tokens/s) of a homogeneous 2P server with an EPYC 9654 is compared to an Nvidia H100 GPU. Roughly speaking, the H100 outperforms the homogeneous server by a factor of 2. Performance in these benchmarks increased with increasing batch size, but the H100 was unable to run the opt-1.3b benchmark with a batch size of 128 due to GPU memory limitations.

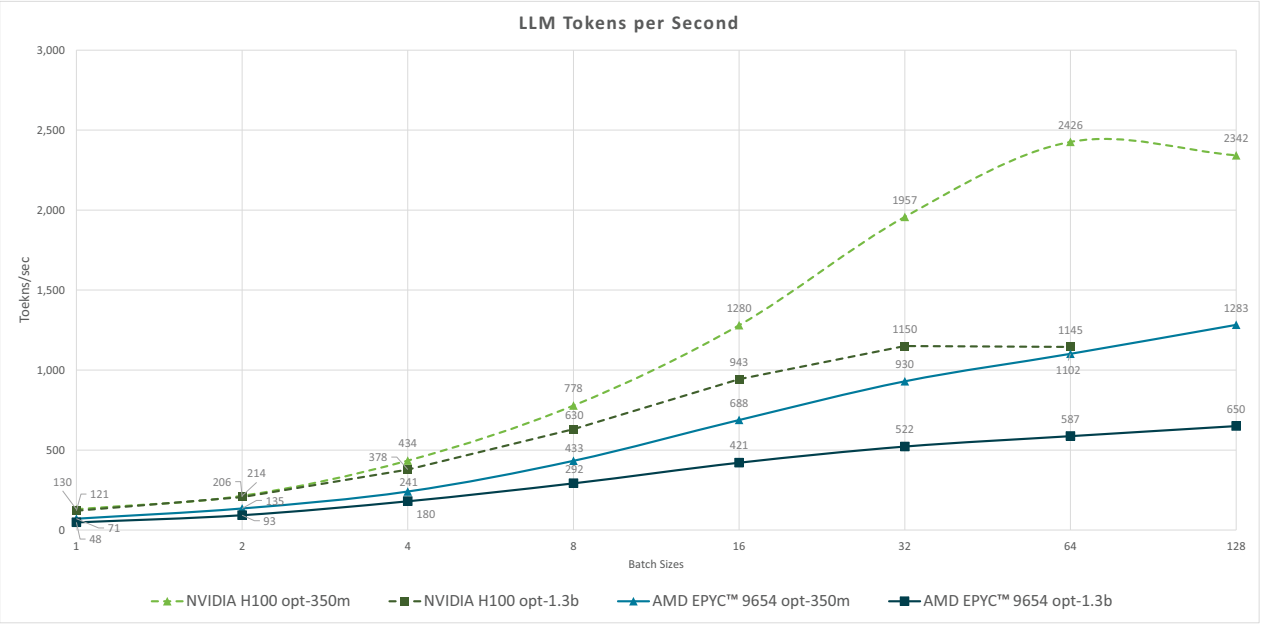


Figure 1: opt-350m and opt-1.3b performance in tokens per second

FIGURE 19. Opt-350m and opt-1.3b performance in tokens per second (figure from [219])

Another interesting test is a transaction processing benchmark that combines various stages of AI problem solving into a single whole. This benchmark, TPC Express AI (TPCx-AI) [**220**], emulates real-world AI scenarios and data science use cases. It is implemented as a pipeline that includes data generation, data management, training, evaluation, and maintenance phases. Here, after machine learning, its accuracy is measured.

TPCx-AI can use different data sets, which provides different scaling levels — SF3, SF10, SF30, SF100, SF300, SF1000 and SF3000, which differ in data volumes. Accordingly, benchmarks up to SF30 are usually carried out on separate servers, and from SF100 and above — in clusters. In addition to the performance value (AIUCpm), the cost-to-performance ratio and energy efficiency are determined (and results in a Hadoop cluster can be selected separately).

In TPCx-AI version 1 to as 27.11.2024, all the highest performance indicators up to the SF1000 level were obtained by Dell using 32-core EPYC Zen 4 (EPYC 9374F were used in single servers, EPYC 9354 in clusters from SF100 to SF1000). For SF3000, the leadership belonged to a cluster from Nettrix with 16 nodes (2P servers with Xeon ICL 8380)[29]. In the new TPCx-AI version 2 to as 8.05.2025 for SF10-SF30, Dell servers with EPYC Zen 5 turned out to be the leaders.

AMD has provided a number of documents showing more high performance of the EPYC 9004 than the Xeon EMR in various AI workloads, including the well-known XGBoost machine learning library. However, this data only shows relative performance and is discussed briefly below in Section 4 when comparing performance to the Xeon EMR or SPR.

2.4.10. *About EPYC 9004 Processor Performance in Cloud-native Workloads*

There are a number of different traditional, non-HPC or non-AI server use areas that are now often used within the cloud, but can of course also be used on physical rather than virtual servers.

[29] https://www.tpc.org/tpcx-ai/results/tpcxai_results5.asp?version=1, accessed 27.11.2024.

AMD has prepared a special guide on how to tuning the EPYC 9004 for cloud computing applications [221]. Since the manufacturer primarily targeted its Bergamo processors at cloud computing, performance data for these processors can be found in AMD community documents focused on this area of application [222, 223].

Unlike the workloads discussed in the previous sections, those discussed here are more likely to not have the entire processor fully utilized at all times, and virtualization may be effective. For example, this includes work with a DBMS.

A common integrated benchmark that evaluates virtualization and a set of typical applications is Broadcom/VMware VMmark [224]. It is well suited for widespread cloud-based data centers, as it combines virtualization with a set of tiles for typical, widely used applications. The available data has different variants, including two nodes with 1P and 2P servers.

Currently, the largest amount of such data available for the processors considered in the review is for VMmark 3.1.1 [225], where, according to the results at the end of 2024, the performance leaders were clearly servers with EPYC 9004, ahead of Xeon SPR and Xeon EMR (servers with ARM processors are not included in these benchmarks). There is still little data for the newer version of VMmark 4, and it relates mainly to the next generation of x86 processors, where the leaders in the benchmarks were Zen 5 [226].

The highest result in the VMmark 3.1.1 benchmark at the time of review writing was shown by a system of two 2P servers from Supermicro with EPYC 9684X, with a performance of 47.7846 tiles. A similar system of two 2P servers from Dell with 64-core EPYC 7773X showed performance two times lower, 23.6424 tiles, which is slightly higher than that of a similar system with Fujitsu servers with 60-core Xeon SPR 8490H (23.3823 tiles), but slightly lower than that of a similar system with Dell servers based on 64-core Xeon EMR 8592+ (25.3428 tiles). Here are presented the maximum results available in [225] for systems with the above-mentioned processors, but the achieved performance, naturally, depends very strongly on other components of these computing systems.

And as more narrowly focused benchmarks, we can specify the performance data of the EPYC 9004 using different relational DBMSs — PostrgeSQL, MariaDB, and MySQL. The classics for such tests, focused on real-time transaction processing (OLTP) and interactive analytical processing (OLAP), are TPC benchmarks[30], which require the precise execution of a large number of specifications, which can require significant financial costs. Accordingly, data appears there less actively, and for systems with the latest processor models — somewhat later. Although the leaders among servers in terms of performance as of 18.12.2024 in the TPC-C, TPC-E, and TPC-H benchmarks are servers with EPYC processors, in TPC-C since 2022 it is the Supremicro 2P server with EPYC Zen 3, but in TPC-E in 2024 the leader was the Lenovo 2P server with EPYC 9554 (clients used a 24-core Xeon 6442Y), and in TPC-H in 2024 the leader was the HPE 2P server with 32-core EPYC 9174F. In the TPCx-V benchmark for a virtualized server platform with a database workload, the 2P server with EPYC Bergamo also leads.

But more often, very close analogs to TPC benchmarks are used. For example, for PostgreSQL there are many results of special PTS benchmarks, which use pgbench benxhmarks of PostgreSQL (close to TPC-B) [**227**], which give the values of latency or transactions per second in the modes "read only" or "read/write", including for variants with different numbers of clients and different scaling factors. For PostgreSQL 17, there is data mainly for Zen 5 and Xeon 6, and for PostgreSQL 16 or 15, there is many data for EPYC 9004.

However, these PTS benchmarks often show poor performance scaling when "replacing" to processor models with a more large number of cores, and when change from 1P configuration to a 2P. In these benchmarks, the leaders are often not x86 server processors at all, and these results are not considered here. There are PTS benchmark data for MySQL and MariaDB, but there are far fewer results relevant to the topic of this review.

A good option for testing database performance is to use the freely available open source application HammerDB[31], which has developed similar TPC-C and TPC-H benchmarks, TPROC-C and TPROC-H.

[30] https://www.tpc.org/, accessed 17.12.2024.

[31] https://www.hammerdb.com, accessed 18.12.2024.

In particular, they allow you to minimize the load on input-output, and the workload allows you to use the capabilities of processors and memory to a greater extent. HammerDB allows you to work with major commercial and freely available DBMS, and is actively used in scientific research.

AMD has prepared a special document on MariaDB performance [**228**], which shows the performance of a 1P server with a 32-core EPYC 9374F and a 2P server with 16-core Xeon SPR 6444Y in HammerDB's TPROC-C and TPROC-H benchmarks. In TPROC-C, the server with the EPYC 9374F is one and a half times faster than with the Xeon 6444Y, and in TPROC-H, their performance is close. At the same time, the cost of the EPYC 9374F at the end of 2023 was $4850, and two Xeon 6444Y processors was $7244 [**228**]. But in 2024, the price of Xeon 6444Y processors decreased (to $3034 for two processors), while that of EPYC 9374F did not change (data as of 17.12.2024 from [**49**]).

The benchmarks by Dell using MariaDB and mysqlslap tools from MySQL for 2P and 1P servers with EPYC 9654/9654P, as well as a 1P server with EPYC 9354P, are presented in [**162**]. There, the dependence of performance and energy efficiency on memory capacity and configuration is also investigated for the server with EPYC 9354P. By the way, [**162**] also contains data on the performance of the Apache HTTP server on 2P and 1P servers with EPYC 9654/9654P.

For the MySQL DBMS, in [**148**, **196**] AMD provided data on the increased performance of a 2P server with 96-core EPYC 9654 compared to a 2P server with 40-core Xeon ICL 8380 by 2.4 times in TPROC-H and 2.7 times in TPROC-C. And in [**165**, **223**, **229**] AMD indicated higher performance in TPROC-C with this DBMS of a 2P server with 128-core EPYC 9754 over two times compared to 2P servers with ARM Ampere Altra Max M128-30 processors and 56/60-core Xeon SPR/EMR 8480+/8490H.

Finally, we will provide data for another important workload — in the field of enterprise resource planning (ERP), performance in 2-tier SAP SD. For a 2P server with EPYC 9654, it was 148 thousand SAPS, for a 2P server with 64-core EPYC 7763 — 75 thousand SAPS, and for a 2P server with 40-core Xeon ICL 8380 — 48 thousand SAPS (SAPS means SAP Application Performance Standard) [**148**].

The performance data above is primarily for the high-end EPYC 9004 models. Information for the mid-range EPYC 9004 models is presented below when compared to Xeon SPR in Section 4.3. We do not cover the performance data for the EPYC 8004 series in this review, which AMD is targeting at Intelligent Edge workloads, data analytics, and decision engineering at the place where the data is generated. Some performance data for this application area is available in [**230**].

In conclusion of the entire Section 2 on server processors of the Zen 4 architecture, it is worth noting the efficiency (including for cost indicators) of the chiplet-based hierarchy of microarchitecture construction and, accordingly, concrete models of EPYC processors. The data provided show significant advantages in terms of performance compared to the previous generation of Zen 3, but usually also in relation to Xeon SPR and Xeon EMR processors. The main comparative analysis of the performance of EPYC 9004 compared to Xeon SPR, Xeon Max and Xeon EMR is carried out further after the analysis of their microarchitecture in a separate Section 4. There are also absolute indicators of the data of the "mass" performance benchmarks SPEC and PTS/OpenBenchmarking.

3. 4th and 5th Generation Intel Xeon Scalable Processors

This section covers the 4th generation (Xeon SPR) and 5th generation (Xeon EMR) Xeon Scalable processors, as well as Xeon SPR with HBM memory (Xeon Max), classified as the conditional 4th generation x86. A general illustration of their high-end models is provided in Table 23 and Table 24, which also provides data on EPYC Zen 4, Zen 3 and 3rd generation Xeon ICL processors. An important but common indicator for all modern server x86 processors — support for the SMT mode (called Hyper-Threading by Intel) is not provided in this table.

The higher core count in the high-end EPYC 9004 models is clearly due to the use of more advanced TSMC technology, which often resulted in advantages in important for performancel indicators. The lower core count in the Xeon can, accordingly, make other lower indicators sufficient for efficient operation, and their mutual consistency in the processor is important here, which allows for maximizing the actual achievable performance.

Table 23. Specifications of modern server x86 processors (high level models)

	Intel Xeon processors				AMD EPYC processors		
Codename	Ice Lake-SP	Sapphire Rapids-SP		Emerald Rapids-SP	Milan	Bergamo	Genoa
Processor family or architecture	3rd Generation Xeon Scalable	4th Gene-ration Xeon Scalable	Xeon Max (Xeon SPR c HBM)	5th Gene-ration Xeon Scalable	Zen 3	Zen 4	Zen 4
Processor model series	Xeon 8300	Xeon 8400	Xeon 9400	Xeon 8500	EPYC 7003	EPYC 9004	
Top-model	Xeon 8380	Xeon 8490H	Xeon Max 9480	Xeon 8592+	EPYc 7763	EPYC 9754	EPYC 9654
Number of cores	40	60	56	64		128	96
Base frequency, GHz	2.3	1.9			2.45		2.4
Turbo frequency, GHz	3.4	3.5		3.9	3.5	3.1	3.7
Turbo frequency of all cores, GHz	—	2.9	2.6	2.9	—	3.1	3.35
FLOPS/cycle per core[1]	32				16	24	24
Peak performance, GFLOPS[2]	2944	3648	3405	3891	2509	6912	5530
Technology, nm	10				7	5	
Price, $	9359[3]	17000	12980	11600	7890	11900	11805

[1] For FP64;
[2] Calculated for base core frequency;
[3] Price as of 4.08.2024 from [231]

Table 24. Specifications of modern server x86 processors (high level models)

	Intel Xeon processors				AMD EPYC processors		
Codename	Ice Lake-SP	Sapphire Rapids-SP		Emerald Rapids-SP	Milan	Bergamo	Genoa
TDP, W	270	350			280	360[2]	
Interconnect of processors in server	UPI: 3×20	UPI:4×24		UPI:4×24	IF:4		
Maximum interconnect bandwidth[1]	11.2 GT/s 67.2 GB/s[3]	16 GT/s 192 GB/s	16 GT/s 192 GB/s	20 GT/s 240 GB/s	128 GB/s[1]	256 GB/s	
L1 cache (per core)	64 KB	80 KB			64 KB		
L2 cache (per core)	1 MB	2 MB			512 KB	1 MB	
L3	60 MB	112.5 MB	112.5 MB	320 MB	256 MB	256 MB	384 MB (32 MB/CCD)
L3 c 3D V-cache	no				768 MB	Up to 1152 MB	
DDR type	DDR4-3200×8	DDR5-4800×8		DDR5-5600×8	DDR4-3200×8	DDR5-4800×12	DDR5-4800×12 Up to 6TB
DDR bandwidth	205 GB/s	307 GB/s	307 GB/s		205 GB/s	461 GB/s	
HBM-memory	no		64 GB	no			
Peak HBM bandwidth	—		1638 GB/s	—			
PCIe: version, number of lanes	4.0, 64	5.0, 80			4.0, 128	5.0, 128	

Basic data and prices as of 27.04.2025 from the database [49].

[1] Bandwidth per processor

[2] Configurable from 320 to 400W

[3] Data from [232]

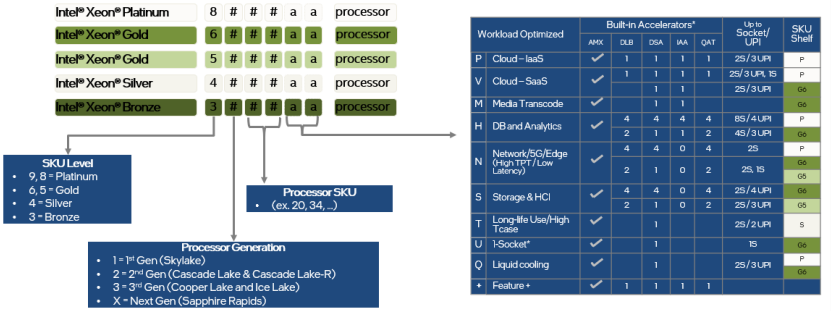
[4] According to [23].

One of the most important HPC enhancements in the release of the Xeon Scalable processors is the introduction of the AVX-512 ISA extension, which includes FMA operations on 512-bit vectors. AVX-512 was already discussed a little earlier, in Section 2.1 on the Zen 4 cores, although only a subset of AVX-512 was implemented there (for more details on AVX-512, see, for example, [**110**, **233**]). Of course, this greatly contributed to the achievement of higher performance of Xeon cores in HPC compared to AMD server processor cores for many years. Further discussion in this subsection is based on the data in [**233**].

The various generations of scalable Xeon processors included many actual improvements that became common to subsequent generations and are also used in the processors considered in this review. From the first generation of scalable Xeon processors (Skylake-SP), it was assumed that the number of processor cores could be increased in the future, which required switching to a different version of their interconnection, the mesh topology of which began to be used. Such a mesh is shown in Figure 22 below. From the very beginning, the focus was also on the possibility of building servers with a number of sockets from 2 to 8.

In accordance with the possibility of producing a very large number of different Xeon models with different numbers of processor cores and different permissible numbers of processors per server, scalable Xeons received additional brand clarifications of the names (in order of increasing their computing capabilities, performance and price): Bronze, Silver, Gold and Platinum. The first two provided the ability to work in no more than a 2P-version of servers, and Gold and Platinum — also in servers with 4 sockets. And only Platinum could be used in 8-processor servers [**233**]. But the number of a specific Xeon model unambiguously assigns it to a certain brand variant. This can be seen in Figure 20.

The general model numbering system in the form of four decimal digits LGmn, where L denotes the processor level (Platinum, Gold, Silver, Bronze), G is the processor generation number (mn numbers concrete SKUs) is expanded by adding a number of suffixes (see Figure 20). The ability to use different accelerators and their combinations in different SKUs adds clarifications for specifying SKUs. This is aimed at achieving the efficiency of using different SKUs in different narrow areas of application. L=8 or 9 means Platinum, L=6 — Gold [**235**]. G=4 corresponds to the 4th generation of scalable Xeon processors (Xeon SPR), G=5 — the 5th generation (Xeon EMR).



	Workload Optimized	Built-in Accelerators*				Up to Socket/UPI	SKU Shelf
		AMX	DLB	DSA	IAA	QAT		
P	Cloud – IaaS	✓	1	1	1	1	2S / 3 UPI	P
V	Cloud – SaaS	✓	1	1	1	1	2S / 3 UPI, 1S	P
				1	1		2S / 3 UPI	G6
M	Media Transcode	✓		1	1			G6
H	DB and Analytics	✓	4	4	4	4	8S / 4 UPI	P
			2	1	1	2	4S / 3 UPI	G6
N	Network/5G/Edge (High TPT / Low Latency)	✓	4	4	0	4	2S	P
			2	1	0	2	2S, 1S	G6
								G5
S	Storage & HCI	✓	4	4	0	4	2S / 4 UPI	G6
			2	1	0	2	2S / 3 UPI	G5
T	Long-life Use/High Tcase	✓		1			2S / 2 UPI	S
U	1-Socket*	✓		1			1S	G6
Q	Liquid cooling	✓		1			2S / 3 UPI	P
								G6
+	Feature +	✓	1	1	1	1		

FIGURE 20. Xeon SPR and Xeon EMR numbering system (figure from [234])

On supercomputers and in HPC clusters, the highest-level processors, Platinum (which are mainly discussed in the performance data below), are most often used, although Xeon Gold is also widely used. Suffixes that correspond to the orientation specifically to HPC tasks are practically absent among these processors.

Although the list of suffixes includes M, which according to [166] is oriented towards media, AI and HPC, [235] does not mention HPC. The author is not aware of any data confirming the active use of such processors for AI or HPC. More often, one could see the use of the following suffixes in these areas and the corresponding benchmarks:

H — for working with a database or suffixes for orientation towards cloud technologies,

Y — for IaaS,

P — for IaaS with increased frequencies,

V — for SaaS).

One can also pay attention to processors with the Q suffix, oriented towards liquid cooling (and correspondingly higher frequencies), which may be relevant for some high-performance servers. However, one should look at the real performance of the models (see Section 4.1 below).

The + symbol at the end of the model number means that it has one each of the DSA, DLB, QAT and IAA accelerators enabled (see Section 3.1.3 for more information about them). A full description of the suffixes is available in [235].

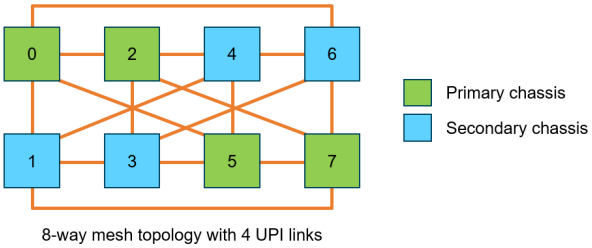


Figure 21. Interconnect topology in an 8-processor Lenovo server with Xeon SPR (figure from [236])

But let's return to the general characteristics of the microarchitecture of these generations of processors. Each core and the corresponding slice of L3 cache (Intel prefers to use the term last-level cache, LLC) has a special CHA (Caching and Home Agent) unit. It maps access addresses to a specific LLC bank, memory controller, or I/O subsystem, and provides information about the necessary routing to reach the destination using the mesh interconnect. The "distributed" construction of the entire CHA in the processor allows for scaling as the number of cores used increases.

This mechanism allows for reduced latency in data exchange between cores and the L3 cache, memory, and I/O system compared to the need to traverse a ring interconnect and possibly a switch between the two rings in the previously used Xeon processor core interconnect topology [233].

The servers uses UPI (Ultra Path Interconnect) as a new interconnect for communication between scalable Xeon processors. Multiple bidirectional UPI channels can be used to connect to other processors. UPI uses the directory-based home snoop coherence protocol (see [233] for quite detailed information). As an example, Figure 21 illustrates the UPI interconnect topology for an 8-socket server.

This topology applies to Xeon SPR processors, which have by four UPI channels, while Xeon Scalable processors with fewer UPIs per processor (such as Xeon Skylake) have a different topology. Xeon EMR does not support building servers with more than two sockets.

Another hot new feature of the Xeon Scalable Processors is sub-NUMA clustering (SNC). SNC creates multiple localization domains in the processor (here, according to [233], there are two), mapping the addresses of one of the local memory controllers to one half of the LLC fragments closer to this memory controller. And the addresses mapped on the other memory

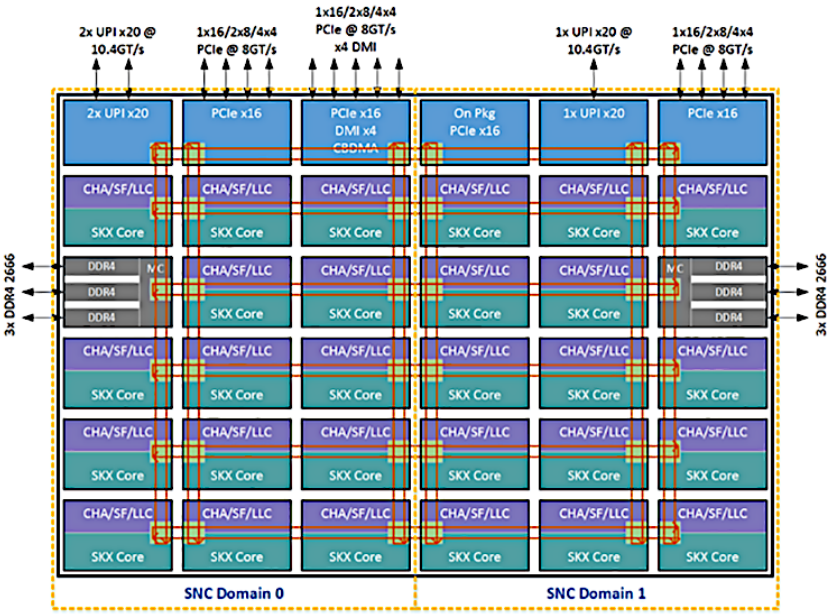


FIGURE 22. SNC clusters in Xeon Scalable processors (figure from [233])

controller are bound to LLC fragments in the other half of the processor. Due to this, processes running on cores in one of the SNC domains use memory at the memory controller in the same SNC domain and get lower LLC and memory latency compared to the latency when accessing outside this SNC domain. To operate in SNC mode, the memory must be filled symmetrically.

The SNC has a unique location for each address in the LLC. The address localization in the LLC for each SNC domain applies only to addresses mapped to memory controllers in the same socket.

This configuration with two domains SNC0 and SNC1 is shown in Figure 22.

The LLC cache in Xeon Scalable processors has become non-inclusive (previously it was inclusive). For more details on the change in cache hierarchy, see [233]

Among the common hardware features used by several generations of Xeon Scalable processors, in particular Xeon SPR, Xeon Max, and Xeon

EMR, it is useful to point out here the virtualization features, including Scalable I/O Virtualization [237]. A detailed analysis of these features is not provided here. [233] lists a many more new features of Scalable Xeon processors that are not mentioned here. We limit ourselves here to those common features that will be discussed later for the concrete Xeon generations under review.

3.1. 4th Generation Xeon Scalable Processors (Xeon SPR)

While the Xeon ICL was a monolithic processor, the Xeon SPR uses chiplets. This processor contains 4 tiles connected using 2.5D -Embedded Multi-die Interconnect Bridges (EMIB) technology [238]. What each tile contains is described further in Section 3.1.3.

The main scientific publications of Intel employees concerning the Xeon SPR microarchitecture are [238] and [239]. [238] mainly considers a lower level, close to semiconductor technology. In this section, we present only information closer to the microarchitectural level. We will only note here that Xeon SPR uses special methods of a higher level than just a semiconductor level, allowing to eliminate defects and increase the yield of processors suitable for operation [238].

There is many interesting information for microarchitecture analysis in [239]. But given the very rapid improvements that Intel has made to its processors, the analysis in the rest of the review relies more on newer and more detailed technical data from Intel's website.

The maximum number of cores in Xeon SPR compared to Xeon ICL and the L2 cache capacity have increased by one and a half times; the L3 cache capacity has also increased significantly (see Table 25 below).

As noted above, Intel demonstrates the orientation of its Xeon models by their suffix. For example, the N suffix offers network-optimized Xeon SPR [240]. They come in Medium Core Count (MCC) and eXtreme Core Count (XCC) configurations. MCC processors have 24-32 cores per socket at 165-205 W, while XCC processors have 52 cores per socket at 300 W.

The striking feature of Xeon SPR is that they are also equipped with special accelerators developed by Intel, and the company offers the possibility of using different accelerators for processors aimed at different areas of application.

Accelerators are designed to improve performance in specific application areas, which contributes to the formation of numerous different processor models optimized for clear and often quite narrow areas of work. Some of the accelerators that can be used by Xeon SPR processors are replacements or upgrades of those previously used by Intel. At the same time, different Xeon models can have not one, but up to four accelerators, or they can all be disabled.

To work with accelerators, it is necessary to use special (in addition to drivers) Intel software, often implemented as libraries. An obvious disadvantage of using accelerators is the need to modernize the application code and the emergence of problems with portability of such code to other platforms. In accordance with general orientation of the review, accelerators are not considered in detail in the review, but brief information about them is given below along with very few references to the first works that appeared, demonstrating the performance achieved with their use.

Another feature of Xeon SPR is support for working with Intel Optane persistent memory. In particular, the focus on possible work with Optane is also present in the Intel In-Memory Analytics Accelerator (IAA) accelerators. However, due to economic inefficiency, Optane production was discontinued in 2023[32], and this hardware is not considered in the review.

3.1.1. *Xeon SPR Core Microarchitecture*

The core microarchitecture used in Xeon SPR is Golden Cove, the successor to Ice Lake's Sunny Cove microarchitecture [**241**], and is in general shared with Xeon Max cores.

A block diagram illustrating the Golden Cove microarchitecture is given in [**239**]. Below in Figure 23 is a block diagram very similar in content, the functionality of most of the blocks of which is clear from their names. We only note that P0, ..., P11 are numbered ports.

All the analysis of the Golden Cove microarchitecture here is based on [**241**], and the improvements are relative to the Ice Lake-SP (Sunny Cove) core microarchitecture. These improvements are mostly quantitative, such as increasing the execution "width" (number of ports in different blocks) or block capacity.

[32] https://www.techpowerup.com/310819/intel-optane-still-not-dead-orders-expanded-by-another-quarter, accessed 5.08.2024.

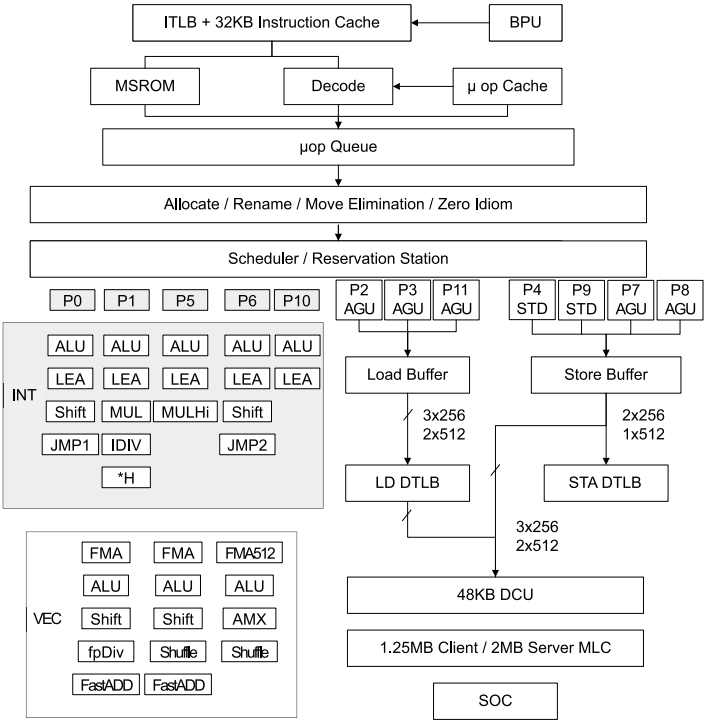


Figure 23. General functionality of the pipelines of cores with the Golden Cove microarchitecture (figure from [241])

In the front-end part of Golden Cove, which is very important for the performance of modern processors with OoO support, the bandwidth of the decoding pipeline has doubled to 32 bytes/cycle, and the number of decoders has increased from 4 to 6, allowing 6 decoded instructions per clock. The Instruction Decode Queue (IDQ) in Golden Cove can store 144 micro-ops per logical processor in single-thread mode (or 72 micro-ops per thread with SMT enabled). The micro-op cache has increased in size, and its bandwidth has increased to 8 micro-ops per clock.

Branch prediction has also been improved, especially for code-heavy workloads. ITLB capacity has been doubled to 256 entries for 4 KB pages, and 32 entries for 2/4 MB pages. Outstanding page miss (miss under miss)

handling has been improved. Maximum load bandwidth has been increased from 2 to 3 loads/cycle.

The L2 cache size in the core has been increased to 2 MB. In Golden Cove also increased the number of ports for execution units from 10 to 12 (see Figure 23), and some of these ports have expanded functionality. There are other improvements in the core microarchitecture, for example in the retirement — see [**241**] for more details.

A direct comparison of such quantitative indicators of the Golden Cove and Zen 4 microarchitectures can be made — using, for example, the data in Table 4 and those available in other sources, references to which are given in Section 2.1 above. In some of these indicators, Golden Cove is better. But, as noted above, manufacturers often evaluate the efficiency of their microarchitectures integrally, by the achieved IPC value. Since the core microarchitecture is often used not only in server processors, but also in PCs, this value is more actual there, given the larger number of sequential applications used.

IPC depends on the application or benchmark used to calculate it, and comparisons of some average estimates are limited in essence, although companies often refer to the IPC progress in their cores. Usually, this refers to IPC when running SPECcpu benchmarks, although the results for SPECint and SPECfp will differ (see, for example, [**242**]). The difference in achieved IPC between Golden Cove, Raptor Cove (microarchitectures of Xeon EMR cores) and Zen 4 is no more than a few percent. Therefore, for servers with the processors considered in the review, a comparative analysis of the IPC of cores is not so actual, and a direct comparison of individual blocks of their microarchitectures is not carried out here.

However, we point to a thorough comparison of Nvidia Grace (Neoverse V2), Intel Golden Cove, and AMD Genoa microarchitectures in [**13**]. Some potential advantages of the Grace microarchitecture are narked there, but the comparison of the achieved performance data is more actual. It is noted there that Genoa achieves 78% of its theoretical peak memory bandwidth, while the Grace superchip and Xeon SPR achieve 87% and 90%, respectively. However, according to AMD data in [**121**], the 2P server with EPYC 9654 achieves a maximum of over 85% of the peak bandwidth (see above in Section 2.4.2), which is very close to the value for Grace.

In [**13**], concrete models were compared: 72-core Grace with 52-core Xeon 8470 and 96-core EPYC 9654. It also provides interesting data on the real reduction of frequencies (using turbo mode in x86) under high

computing loads, including using AVX-512 with different numbers of cores. At the same time, the frequency of the Xeon 8470 cores decreases much more than that of the EPYC 9654, both with and without AVX-512, while the frequency of Grace is almost unchanged.

It should also be noted here that Xeon SPR cores provide increased performance thanks to a programmable power management controller [238].

The Grace processor does not support 512-bit vector arithmetic at all. [13] provides estimated peak performance data for the maximum possible frequencies and those obtained when they are reduced under a workload on the maximum number of cores. But even with this calculation, Grace lags far behind the EPYC 9654 in peak FP64 performance. This is combined with Grace's lower TDP and higher supported bandwidth of the more expensive LPDDR5X memory used to work with Grace, which corresponds to the focus on using Grace only in servers with Nvidia GPUs. And without real data on the achieved application performance, further comparison becomes irrelevant.

In server processors, in addition to IPC, there is a large number of other important parameters — in addition to the permissible clock frequencies and the number of cores, this includes, for example, power consumption and costs, which also depend on the technology used. But the most important are the performance data (for individual cores — in sequentially executed benchmarks and applications) — this is discussed further in Section 4.1.

3.1.2. *ISA Extensions in Xeon SPR Cores*

Naturally, there are many sources of relevant information on the Intel website. Given the focus on improvements in the Xeon SPR hardware implementation, a brief summary of the ISA extension is available in [243].

A complete guide to the modern ISA is available in [244] (see also [241]). The most important instruction set extension, in our view, is AMX. It is focused on AI tasks (deep learning and AI inference) and is used starting with Xeon SPR — i.e. also in Xeon EMR and in Xeon 6 P-cores. In our view, AMX is very important and has large prospects for further development. AMX can be considered a kind of new programming paradigm in x86.

AMX is not just an instruction set extension; there is a dedicated hardware block to implement it. AMX can be classified as an accelerator — but like AVX-512 (unlike the other accelerators mentioned above), the AMX unit is present in every Xeon SPR core. And the most important

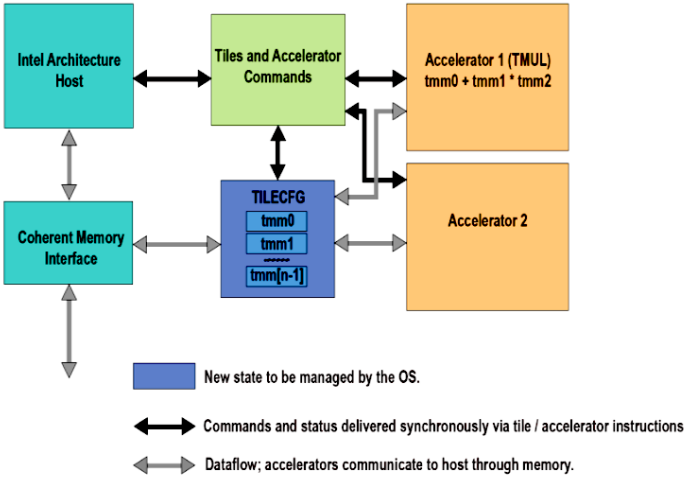


FIGURE 24. Conceptual Diagram Of The Intel AMX Architecture (figure from [243]).

information about AMX, of course, pertains to the instruction set. The conceptual block diagram related to the AMX implementation is shown in Figure 24, and is used here to illustrate the relevant part of the ISA.

Details of AMX, on which the analysis below is based, are available in [245]. AMX is designed to handle BF16, FP16, and INT8 data matrices relevant to AI tasks, and includes two main components. These are a set of two-dimensional registers (tiles), which essentially represent subarrays of a larger two-dimensional memory image, and an accelerator capable of operating on these tiles. The first implementation of such an accelerator is called TMUL (tile matrix multiply unit) [245]. AMX is an extensible architecture, allowing for the addition of new accelerators or data formats, or enhancements to TMUL to increase performance.

AMX lists a palette of options for the programmer to specify how the tiles can be programmed. There are currently two palettes, the primary one for AMX use being palette 1. It contains a set of 8 tiles — registers named TMM0...TMM7. Each tile has a maximum size of 16 rows of 64 bytes (1 KB), but the programmer can configure each tile to smaller sizes to suit the algorithm he is using [245]. This naturally corresponds to a 16×64 matrix with INT8 format or a 16×32 matrix with BF16 format. Information about the maximum number of tiles supported and the maximum tile size for the particular hardware being used can be obtained via the CPUID instruction.

The palette number (palette_id) and tile sizes are metadata stored inside another special control register (TILECFG). Since execution is driven by metadata, the existing Intel AMX binary can take advantage of the larger storage sizes and more efficient TMUL blocks by choosing the most powerful palette specified by the CPUID and adjusting the cycle and pointer accordingly. TILECFG is programmed with the LDTILECFG instruction. AMX initially included 12 instructions, then their number increased to 17, in part due to the expansion of the data formats used. These instructions perform general setup, tile load/store, and the actual matrix multiplication operations on the tiles. Difficult instructions such as matrix operations require many clock cycles to execute in TMUL [**245**]. Data about number of cycles and latencies for AMX instructions is available in [**246**].

AMX also uses terms similar to those used for regular accelerators, such as GPUs. For example, the part of the processor and the memory it uses is called the host (see Figure 24) — this term is used when servers have regular accelerators, including GPUs. AMX is initially aimed at future improvements and scalability. In this figure, two accelerators are indicated, although Xeon SPR has one TMUL. Future improvements to TMUL and/or increase the number of accelerators are possible [**245**].

AMX instructions are synchronous with the regular x86 instruction stream, and tile load/store instructions provide coherence with respect to host memory. The host can use any of its resources (e.g., AVX-512) in parallel with AMX [**245**].

It should be noted that the ability to work with AMX is provided only by fairly new versions of Linux kernels (from 5.16.20 and higher). This means that the kernel versions supplied in currently commonly used distributions (for example, RHEL 9 and all distributions compatible with it, SLES 15.4 and the corresponding OpenSUSE, Ubuntu 22.04) are not suitable for AMX. But AMX is supported by kernels in SLES 15 SP6, Ubuntu 22.10 and higher, and Fedora from version 36 and higher.

The available performance data on AMX for AI tasks on Xeon SPR processors is discussed further in Section 4. About the possible security violations in modern processors it was already said above. According to [**247**], AMX can also be used for this purpose.

The Xeon SPR has many extensions to the ISA. Some of the new instructions are centered around the Accelerator Interface Architecture (AiA). They are built into the ISA and are an enhancement to x86 processors designed to optimize the movement of data from accelerators to host services [**243**].

Other extensions noted in [**243**] include new commands to support virtualization and security. For details, see [**245**, **246**].

3.1.3. *Xeon SPR Processor Microarchitecture*

We begin our analysis of the Xeon SPR microarchitecture with the general for "around 4th generation" Xeon processors Table 25. It complements the data in Table 23 and Table 24 and generally characterizes the changes in the quantitative indicators of these processors, which largely determine the progress achieved in the corresponding generations. In the text below and in Table 25, more attention is paid to bandwidth due to their importance for performance.

As for the amount of addressable physical and virtual memory, for traditional dual-socket Xeon processors use 52 and 57 bits, respectively, the same as the EPYC 9004. This also applies to Xeon SPR in 4- and 8-socket configurations (in the previous generation, Cooper Lake-SP, the addressable memory was smaller).

The maximum theoretical bandwidth (B, GB/s) of the memory and interconnect between processors is calculated from the data in Table 25 using the usual formula

$$B = N \times S \times W/8,$$

where N is the number of channels (memory or UPI), S is the transfer rate (GT/s for UPI or one thousandth of MT/s for DDR), W is the channel width in bits (for example, 24 for UPI in Xeon SPR and EMR, and 64 for DDR4 or DDR5).

Based on the data in Tables 23 and 24, the progress in Xeon SPR (this also largely applies to Xeon Max, see Section 3.2 was primarily manifested in a significant increase in the number of cores, for which Intel's improved technology was used. However, the lag in the number of cores from AMD EPYC, as well as in relation to the semiconductor technology used to produce EPYC, remained.

The L2 and L3 cache capacities have also increased. The transition from DDR4 to DDR5-4800 has been implemented. Given the slower growth of memory bandwidth compared to processor performance, these improvements in the memory hierarchy are extremely important. The transition to PCIe 5.0 support has also been made, and the number of PCIe lanes has been significantly increased.

Table 25. Xeon 3rd, 4th and 5th Generation Scalable processors (SP) specifications

	3rd Generation SP		4th Generation SP Xeon SPR	Xeon Max	5th Generation SP Xeon EMR
	Xeon ICL	Xeon Cooper Lake-SP			
Number of Sockets	1, 2	4, 8	1,2,4,8[1]	1,2	
Technology, nm	10	14	10		
Max number of cores per CPU	40	28	60	56	64
I-Cache L1/D-cache L1	32 KB/48 KB	32 KB/32 KB	32 KB/48 KB		
Cache L2 (private per core)	1.25 MB	1 MB	2 MB		
Max Cache L3	60 MB	38.5 MB	112.5 MB		320 MB
Max TDP	270 W	250 W	350 W		
Physical/Virtual address bits	52/57	46/48	52/57		
Number of Memory controllers /Sub-Numa clusters	4/2	2/2	4/4		4/2
Memory type	DDR4		DDR5		
Max MT/s	3200 (2DPC)	3200 (1DPC) 2933 (2DPC)	4800 (1DPC) 4400 (2DPC)		5600 (1DPC) 4800 (2DPC)
Max number of memory channels (per CPU)	8	6	8		
HBM2E memory size	—			64 GB	—
Max number of UPI links (per CPU)	3×20	6×20	4×24		
Max UPI speed	11.2 GT/s	10.4 GT/s	16 GT/s		20 GT/s
PCIe supported	PCIe v4		PCIe v5		
Max PCI lanes	64	48	80		

[1] > 8 via xNC support (see in Section 3.1.4). Data from [49, 50, 238, 243, 248].

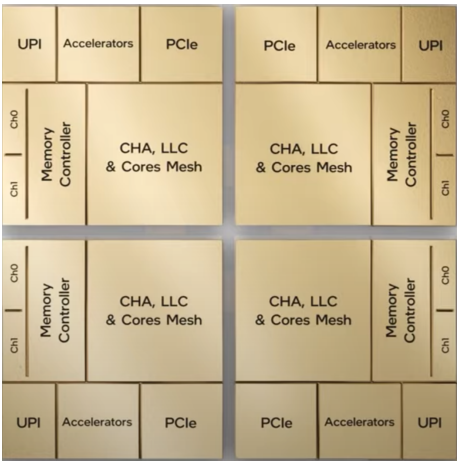


FIGURE 25. Main blocks of the Xeon SPR microarchitecture (figure from [253], corresponds to the one presented by Intel in [239]).

The overall block diagram illustrating the Xeon SPR microarchitecture is shown in Figure 25. All blocks in this figure are clear from their names, and Ch0 and Ch1 at the edges of the figure denote memory channels.

The Xeon SPR is the first to use the Intel EMIB interconnect. It overcomes the single-mesh limitation and provides more efficient scaling than monolithic designs. EMIB is also used by Intel for other hardware, such as FPGAs. EMIB has long been well known from a number of publications [249–252], and the Xeon SPR has been improved accordingly [238].

Figure 25 shows the presence of up to 4 chiplets in different Xeon SPR models. Their use, as can be seen from this figure, means the presence of NUMA already within one processor, since each chiplet has its own near memory and its corresponding Ch0/Ch1 controllers. In fact, such possible localization covers not only memory, but also the LLC cache.

Intel uses the term "unified memory access (UMA) domain" for the memory of Xeon SPR processors [243]. It provides a single address space that interliveas between all memory controllers. However, NUMA clusters (SNC), which were already discussed above at the beginning of Section 3, can be formed from common to the entire processor blocks of memory and chiplets close to them. Taking NUMA into account by using SNC domains

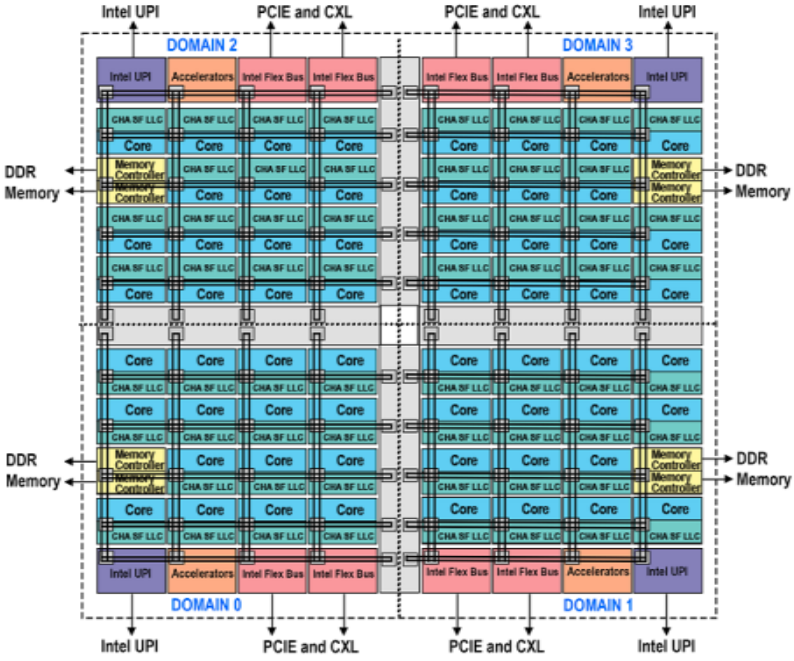


Figure 26. Block diagram of Xeon SPR with four SNC domains (figure from [243]).

allows for the efficient use of lower access latencies to the locally placed part of the LLC and nearby memory. According to Figure 25, the processor can be logically divided into two (SNC2) or four clusters (SNC4).

SNC gives a unique location for each address in the LLC, and it is not duplicated across LLC domains. The localization of addresses in the LLC for each SNC domain applies only to addresses mapped to memory controllers in the same socket. All addresses mapped to remote socket memory are distributed evenly across all LLC banks, regardless of SNC mode. When using SNC mode, each core has access to the processor's entire LLC as usual.

At the partitioning to the four SNC it turns out the "finest" NUMA accounting, shown in Figure 26. Figure 26 is more detailed than Figure 25 and also shows the mesh used as the interconnect for the processor cores [243].

Naturally, practical application of SNC may require modernization of the text of the software using SNC, usually with the use of OpenMP. But Intel has offered two more interesting options for possible consideration of NUMA features without software modifications. Such options with division of the processor into 2 and 4 parts are called Hemisphere and Quadrant modes, respectively.

The processor has access to all cache agents and memory controllers. However, based on the hash function, each home agent automatically provides routing of the cache agent to the nearest memory controller in the same domain. Once the processor accesses a cache agent, the distance is minimized, since it will always communicate with the nearest memory controller [**243**]. As noted above at the beginning of Section 3, cache agents and home agents are combined in CHA blocks. These two modes of operation with hash functions can give higher performance relative to UMA, but, of course, not as good as true SNC clustering. By default, the BIOS is set to the Quadrant mode.

These modes, like true SNC clustering, require symmetrical memory arrangement. If the available memory symmetry is insufficient for quadrant mode, hemisphere mode will be used. And if the memory symmetry is not suitable for this mode either, UMA will work [**243**]. Regardless of the SNC formation options, both memory and LLC remain common to the entire processor, it is simply possible to "prioritize" working with near LLC and memory components.

Accelerators. In Section 3.1.2 it has already been said about the AiA accelerator interface architecture and the most relevant accelerator for AI tasks, the AMX. In addition to it, Xeon SPR can have 4 other types of Intel accelerators — IAA (In-memory Analytics Accelerator), DSA (Data Streaming Accelerator), DLB (Dynamic Load Balancer), and QAT (QuickAssist Technology). The following brief information is based on [**243**].

IAA can speed up performance by using analytics primitives (scanning, filtering, etc.), sparse data compression, and memory tiering. This is important for working with big data and in-memory databases.

Memory tiering involves dividing the memory into different areas to facilitate resource management. Frequently accessed (hot) data can be in an area where the bandwidth for CPU is higher, while less intensively accessed (cold) data can be in an area with slower bandwidth. Hot data can be stored in memory, and cold data on the hard disk.

IAA can have up to four instances per socket that are available to offload data from the processor core. From a software perspective, each instance represents a complex integrated PCIe endpoint. IAA also supports shared virtual memory, allowing the device to operate directly in the virtual address space of applications.

DSA replaces the older Intel Quick Data Technology. It helps offload common data movement operations from processor cores that can be found in storage applications, hypervisors, persistent memory operations (with Intel Optane), networking applications, operating system cache flushing, page zeroing, and memory moves.

DSA has up to four instances per socket, depending on the model. It supports data movement between any platform component, including all compatible attached memory and I/O devices. For example, it supports data movement between the processor cache to main memory, between the processor cache and an add-on card, and between two add-on cards.

Since modern computing systems often abandon the use of HDDs in favor of working with RAM (e.g., in-memory database systems), and the use of CXL memory may become actual, the importance of DSA may potentially increase. DSA operation with CXL is discussed, for example, in [**254**].

Other DSA capabilities arise from the use of dual memory levels (HBM and DDR) in Xeon Max using the Xeon SPR microarchitecture. Large bandwidth gains thanks to the use of DSA in a 2P server with 48-core Xeon Max 9468s, using single or multiple threads, using single or multiple DSAs, and using single or dual sockets, is obtained in [**255**].

According to [**256**], which is cited as an example of how DLB can be used effectively in Intel's Xeon SPR documentation [**243**], DLB is a PCIe device that provides load-balanced and priority-controlled scheduling of events (packets) across processor cores/threads. This accelerator in Xeon SPR is located inside the processor.

In [**256**] the use of DLB is described with the open source library set DPDK (Data Plane Development Kit), although its use is not necessary. Optimization of work with the queue system, for which DLB is applied, is actual from the point of view of [**256**] due to the growing activity of using accelerators. Another possible area of use of DLB is indicated therein as traffic optimization tasks in telecommunications tasks.

QAT is designed to speed up the solution of cryptographic tasks. But if the previous generations of QAT were located in the chipset, then QAT was moved to the processor case. In QAT in Xeon SPR, the speeds of encryption, RSA decryption and data compression were increased.

On the other hand, it is clear that the use of accelerators requires not only the use of Intel drivers, but also special software. And the focus on working with them causes the intolerance of the corresponding code to other hardware platforms.

In our view, potentially more important than the use of accelerators in Xeon SPR (except AMX) may be the presence of CXL 1.1 [**243**] support in these processors, which makes it possible to connect CXL memory as a possible cost-effective way to smooth out the gap between main memory and external memory. Persistent memory (primarily Intel Optane), given Intel's refusal to continue producing it, has not solved this problem.

TDP and frequency change. The maximum TDP values of the Xeon SPR models compared to other generations of Xeon Scalable processors are shown in Table 25, Table 23 and Table 24 compares the TDP of the top-end Xeon 8490H model with the data for the top-end EPYC 9004 models. Although the TDP for EPYC is slightly higher, the core count of the top-end EPYC models is much higher. A certain lag in Intel 7 semiconductor technology compared to TSMC's technology used to manufacture EPYC certainly affects the TDP.

The high-end EPYC Milan and EPYC Genoa models in Table 23 and Table 24 have higher base and turbo frequencies than the Xeon ; the 96-core EPYC 9654 also has a higher all-core turbo frequency than the Xeon 8490H.

It is clear that Intel puts a lot of effort into optimizing power consumption and managing the clock rate applied. Intel Xeon SPR processors introduce new power-optimized C-states C0.1 and C0.2 (they are substates of C0). Like the deeper C-states (C1, C1E, C6), instruction execution is halted as soon as the core enters one of these new C-states. The new C-states have much lower exit latencies, which is important for accelerating common networks, such as those using the DPDK library [**257**].

As for the actual frequency regulation, this is related to P-states. These states in Xeon SPR can provide a reduction in the operating voltage and core frequency and, accordingly, a reduction in power consumption.

Enabling or disabling Intel Turbo Boost technology naturally affects P- and C-states (for more details, see, for example, [**258**]). Turbo Boost is almost always enabled to achieve maximum performance, but it can be disabled to achieve lower, but predictable, performance.

Linux kernels for Xeon SPR have two different drivers. First, there is the traditional acpi-cpufreq driver used by the CPUfreq subsystem. The various governors it has are described above in Section 1.2, and their use in EPYC Zen 4 is described in Section 2.2.

Another driver, intel_pstate, implements two of its own algorithms. The "performance" algorithm implemented by intel_pstate is similar to the CPUfreq algorithm of the same name. And the "powersave" algorithm in intel_pstate is more similar to "schedutil" in CPUfreq. The intel_pstate driver is loaded by default. For vCMTS (virtualized cable modem termination system), it is recommended to work with the "userspace" regulator in the traditional driver, since this allows full control over the frequency setting [**257**].

From the Linux command line, you can monitor and change P-state parameters using the python power.py tool[33] available in the Intel CommsPowerManagement GitHub repository.

In practical terms, an actual improvement in the Xeon SPR is the introduction of a special "Optimized Power Mode" (OPM) in the BIOS, which allows for high performance with a noticeable reduction in power consumption (see, for example, [**259**]). An illustration of the effectiveness of its use is given below in Section 4.1.

Virtualization. The above-discussed features of NUMA application in Xeon SPR are certainly reflected in the actual use of virtualization tools, since this facilitates the natural formation of a virtual machine on each NUMA node.

For example, the well-known business software product SAP HANA, which previously supported the operation of two virtual machines on a 2P server, now supports operation of virtual machines in SNC2 mode, two virtual machines per processor and, accordingly, four per server [**260**].

Intel is generally distinguished by a huge set of areas in which the company achieves success. As an example of the virtualization area, we can mention vRAN (deployment of virtual radio access networks), for which Intel has created special vRAN boost tools, and specialized Xeon Sapphire Rapids EE. However, all this does not fall within the scope of this review.

[33] https://github.com/intel/CommsPowerManagement, accessed 18.11.2025.

Intel's VT-X virtualization technology, including the corresponding instructions in ISA, VMX (virtual-machine extensions), has been around for a long time and has evolved greatly over the years. VT-X has also used the EPT (Extended Page Tables) virtualization technology for a long time. VT-X now has many capabilities. Many of them have analogues in Zen-4 processors, and were briefly discussed above. Here we will focus only on the latest improvements in this area, available in Xeon SPR, which relate to VT-d (Virtualization Technology for Directed I/O), which adds a new level of hardware support for virtualization of input/output devices [**261**].

VT-d uses, in particular, virtualization of IPI (Inter-Processor Interrupts). IPI is part of the Intel APICv interrupt virtualization technology, which is analogous to AMD AVIC. IPI is stably supported only by new versions of Linux kernels (from 5.19 and higher).

Intel, in its main technical document on Xeon SPR [**243**], points to its new Scalable IOV technology introduced in these processors, replacing the previously existing SR-IOV I/O virtualization technology by providing flexible composition of virtual functions using special software for own hardware interfaces.

While SR-IOV implements a full virtual function interface, Scalable IOV instead uses a lightweight interface optimized for fast data path operations for direct guest access. Intel Scalable IOV easily interfaces with PCIe or CXL via process address space identifiers, allowing multiple guest operating systems to simultaneously use PCIe or CXL devices. Intel Scalable IOV also supports all accelerators present in Xeon SPR and any discrete accelerators. However, Scalable IOV requires support from operating systems and virtual machine managers [**243**].

Security. Xeon SPR uses Intel's new Trust Domain Extensions (TDX) technology for hardware-assisted security. EPYC Zen 4 security features were briefly discussed above in Section 2.2.

Research into the security features of all modern processors is an actively developing scientific field. These features are of limited relevance to performance due to their rare use, for example, for HPC. Accordingly, they are not considered in the sections of the review where the performance of the processors analyzed in it is compared. Therefore, it is appropriate to conduct a small comparison of these hardware features of the processors under consideration here.

In the introduction, it was already noted that attacks leading to security breaches are possible for almost all modern superscalar processors. Here we will point out another attack variant for such processors, Downfall, and the corresponding publication [**262**]. As noted in [**262**], Intel claims that Xeon SPR is not affected by this. Preliminary tests of AMD Zen 2 did not reveal any data leakage, but the authors of [**262**] plan to continue research on processors from Intel and other manufacturers.

The hardware security features discussed here are primarily focused on virtualization and cloud computing. Intel, which introduced SGX (Software Guard Extensions) security features for its processors earlier than other manufacturers in 2015, formulated a clear orientation as the basis of its confidential computing paradigm: those who control the platform and those who have data processed on the platform are two separate entities. Whereas typically the owner of the platform has full access to what is processed on the platform [**263**].

Virtualization and its application to cloud technologies naturally raises security issues to a much more important level. As an illustration, we note that, according to IBM, 82% of data leaks were related to data stored in the cloud [**264**].

The term TEE (Trusted Execution Environment) is common to a wide range of security technologies. TEE tools must guarantee confidentiality and protection from unauthorized access, as well as be immune to various types of attacks. It was already noted above that these guarantees are still rather dubious, since data on new security-violating attack variants is constantly appearing. Therefore, TEE tools rather provide only certain types and levels of security. Security tools in ARM processors, SEV in AMD EPYC, and Intel SGX are related to TEE.

The most important hardware metrics for TEE include not only the level of security that can be achieved, but also the level to which these tools can cause a decrease in the achieved performance. As an example of work devoted to their impact on HPC performance, we will point out [**265**]. And it should be immediately noted that the use of both Intel SGX and AMD SEV demonstrated in [**265**] a possible very strong decrease in performance.

Intel SGX is based on isolating and encrypting the parts of code that handle sensitive data in an application. The data is encrypted and decrypted on-the-fly (before being written to or read from system memory) using well-known cryptographic algorithms. A slightly different strategy is to isolate and encrypt the memory of the entire virtual machine, as used in AMD SEV [**266**].

SGX tools have often been considered the best in terms of the level of security achieved, see, for example, [**264**, **267**]. However, there is, for example, the CipherLeaks attack, which is violated the security of both SGX and SEV (see, for example, [**268**]).

As a drawback of using SGX often indicated that it requires modification of application code and places the onus on application developers to write attack-resistant code [**266**], and is therefore considered difficult to use [**264**].

However, the reason for Intel's abandonment of SGX and the transition to TDX (Trust Domain Extensions) technology in Xeon SPR was most likely the limitation of the safe memory size [**264**]. This made SGX unacceptable for HPC tasks [**265**].

AMD, which has used a strategy based on isolating the entire virtual machine's memory in SEV (since 2016), has improved SEV twice already: in 2017 to SEV-ES (Encrypted State) and in 2020 to SEV-SNP (Secure Nested Paging) [**266**]. These technologies were briefly considered earlier in Section 2.2.

The new Intel TDX technology introduced in Xeon SPR, like AMD SEV, allows the deployment of hardware-isolated virtual machines, which in TDX are called Trust Domains (TDs). The further discussion of TDX here is based on [**263**] and the latest (at the time of review preparation) description of TDX [**269**].

TD, a new type of guest virtual machine, extends the capabilities of the previously available VMX (the above-mentioned ISA extension for VT-X) and MKTME (Multi-Key Total Memory Encryption), providing hardware isolation of virtual machines from the hosting environment as well. Even with full control over the host management mechanisms by the VMM/hypervisor, it is not possible to access the private information of virtual machines, as was previously the case with virtualization in cloud hosting. Note that MKTME is built into the memory controller [**38**].

The software component that controls TD is called the TDX module. In Xeon SPR, TDX introduces a new x86 mode, SEcure Arbitration Mode (SEAM), which is used to isolate the TDX module from all software and hardware components that are not included in the TD Trusted Computing Base (TCB). Xeon SPR with TDX restricts access to the SEAM-memory range to software running within the SEAM memory range, and all other software accesses and direct memory access (DMA) from devices to that memory range are prevented.

In the SEAM-memory range provides cryptographic privacy protection using AES in XTS mode with a 128-bit memory encryption key. One of two available memory integrity protection modes can be used. It can be provided either by (the default) a cryptographic protection scheme or by using a logical integrity protection scheme.

The cryptographic integrity scheme uses a 28-bit Message Authentication Code (MAC) based on the SHA-3 cryptographic hash function, and the logical integrity protection scheme is only intended to prevent access to the host/system software (it uses one TD ownership bit).

As a result, the workload is able to use SEAM (regardless of the cloud infrastructure it uses) to provide more secure access to different software and hardware components [**38**, **263**].

Additionally, TDX includes remote attestation capabilities that allow a relying party (such as the workload owner) to establish that a workload is running on a TDX-enabled platform located within a TD before the workload data is provided.

The description here, as well as [**38**, **263**], is based on TDX 1.0. Intel provided detailed data for research testing to Google, Microsoft, and the NCC Group (a well-known cybersecurity company) before Xeon SPR began shipping. They studied TDX for security compliance and provided Intel with the data, which Intel then used to make the necessary improvements.

The most famous of these tests was probably a 9-month study at Google, the detailed results of which were published in [**38**]. It tested 81 attack vectors, which yielded, in particular, about 10 security vulnerabilities. According to [**269**], all of them have been fixed in the Intel Xeon SPRs released, which use the newer version of TDX.

These data, on the one hand, speak of the very high complexity of such implementation of security tools. But Intel received a great advantage as a result of such research. In [**38**], the complexity of the TDX system itself is also noted.

It is also important that TDX quickly became available in well-known Linux distributions, such as RHEL 8.10, SLES 15 SP5 or Ubuntu 24.04 LTS.

Like the other security tools discussed, TDX is focused on virtualization and cloud computing, and cannot address all potential processor security

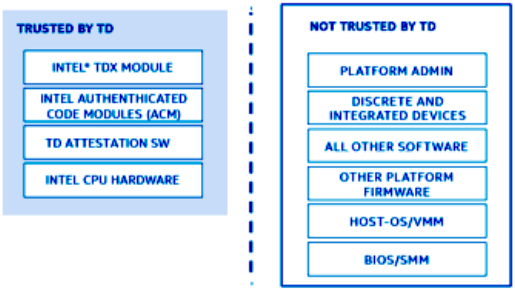


FIGURE 27. Trust Boundaries for TDX (figure from [263])

issues. Figure 27 illustrates which security areas TDX covers and which it does not.

In comparative evaluations of different TEE technologies, it is necessary to take into account such "side" effects as reduced performance, the need for support in the operating system, and the modernization of the codes of applications using TEE. In modern works, new variants for ensuring security are proposed (see, for example, [264, 270]). All this requires further research, and it remains reasonable to simply carefully compare the existing capabilities of TEE technologies, as was done earlier in [267].

3.1.4. *Computing Systems on Xeon SPR*

Before moving from processors to computing systems, we need to talk about the chipset. Although Intel classifies Xeon SPR as SoC, the chipset (C741) for these processors, unlike EPYC 9004, is used. The C741 was already mentioned above in Section 1.1. The presence of this chip slightly increases the cost ($70) and the TDP value (11 W) [45].

Traditional dual-socket Xeon SPR-based systems are offered by all major server manufacturers, and there is nothing particularly new architecturally. We note only that the use of NUMA requires careful configuration with respect to memory module placement. A useful practical guide for dual-socket servers may be Lenovo's recommendations for servers with Xeon SPR and Xeon EMR processors [271].

Supermicro and Lenovo produce servers with 4 or 8 sockets. There are no analogs of such servers with AMD EPYC. It is also fundamentally important that systems with a common memory field containing more than 8 sockets can be built on the basis of Xeon SPR processors — this is possible using xNC (eXternal Node Controller) technology, developed by Intel jointly with Numascale [272].

Famous compute systems that used xNC capabilities were previously the HPE Superdome Flex. Similar systems based on Xeon SPR, the HPE Compute Scale-up 3200 Server, have a modular architecture using 4-socket 5U building blocks. The architecture of these systems, which will ship in 2023, is a combination of the Superdome Flex Server which used xNC and the SuperdomeFlex 280 Server which uses a glueless infrastructure with increased UPI connectivity [**273**].

The Compute Scale-up 3200 system provides scalability from 4 to 16 sockets in 4-socket increments and DDR5 scaling up to 32 TB. The architecture of these servers is described in more detail in [**274**]. They are primarily intended for SAP HANA and in-memory database workloads.

The supercomputers in the June 2024 TOP500 list contain Xeon SPRs in both heterogeneous GPU nodes and homogeneous CPU-only nodes.

The most famous American supercomputer with Xeon SPR, which also contains Nvidia H100 in its nodes, is Microsoft Eagle ND v5 [**275**] — the third-placed Microsoft ND v5 virtual machine from the Azure family, running Ubuntu 22.04. There, 56-core Xeon 8480C are used in the nodes. The C suffix in the SKU name is missing from the general list of suffixes in Figure 20. It is possible that this is a preliminary version of the Xeon 8480+. It uses the same cores with a base clock frequency of 2 GHz, but the maximum allowable temperature is 73 degrees Celsius, while in the Xeon 8480+ it is increased to 79 degrees.

The European supercomputer Leonardo, which ranks 7th on this list, has a so-called "data-centric" part (we will use the term partition for such parts-clusters from now on) with homogeneous nodes based on Xeon 8480+, but it is supplemented by a "booster" partition, where GPUs are used in the nodes [**276**].

The 22nd-placed Mare Nostrum 5 GPP computing system at the renowned Barcelona supercomputer center [**277**] uses homogeneous GPP nodes. There are 3 types of them: regular, with extended memory, and with HBM memory. Regular nodes and nodes with extended memory use two 56-core Xeon 8480+ with 256 GB and 1 TB of DDR5 memory, respectively. Nodes with HBM use 56-core Xeon CPU Max (see Section 3.2 below). MareNostrum 5 GPP uses Lenovo ThinkSystem SD650 V3 as servers.

But in addition to homogeneous nodes, MareNostrum 5 also has heterogeneous nodes, each containing two 40-core Xeon 8460Y+/2.3 GHz and four H100 GPUs. In general, MareNostrum 5, using the InfiniBand NDR200 interconnect, is a pre-exascale supercomputer of the EuroHPC joint venture [**277**].

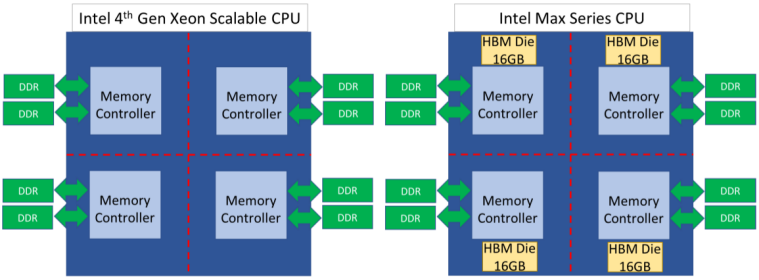


FIGURE 28. Comparison of Xeon SPR and Xeon Max architectures (figure from [280]).

Performance data for computing systems with Xeon SPR processors are discussed further in Section 4, and the next section analyzes a modified family of Xeon SPR processors that adds support for working with HBM memory, including in combination with DDR memory. Accordingly, supercomputers containing Xeon SPR nodes with HBM, such as Crossroads [278], are also discussed further.

3.2. Xeon Max Processors (Xeon SPR with HBM)

At first, Xeon Max processors were generally called Xeon Sapphire Rapids with HBM memory in publications. The main difference between Xeon Max and Xeon SPR is the presence of one HBM2e high-speed memory controller in each of the four chiplets in addition to the usual DDR5 controllers. The maximum possible number of cores in Xeon Max has decreased by four, to 56.

HBM memory is known to consist of multiple DRAM stacks and uses a large bus width to provide high bandwidth. Although HBM memory is commonly used in modern GPUs [2], it is also used in the famous Fujitsu A64FX ARM processors [7]. HBM memory is expensive and its capacity in the GPU or A64FX is fixed, as also is the case with Xeon Max.

The Xeon Max has 4 HBM dies — stacks containing 8 DRAM modules each. By one HBM die with a capacity of 16 GB is connected to each of the four HBM memory controllers [280] (see Figure 28). Accordingly, the total HBM capacity in each Xeon Max model is 64 GB. With a bus width of 1024 bits and a speed of 3.2 GT/s, four stacks can provide a peak bandwidth of 1638.4 GB/s. However, due to the peculiarities of the arrangement of individual processor cores in the Xeon Max mesh (see Figure 26), their

Table 26. Xeon Max Model Specifications SKU

number	Number of Cores	Base Frequency	L3 cache size	Max Number of UPI Links	Price
9462	32	2.7 GHz	75 MB	3	$7995
9460	40	2.2 GHz	97.5 MB	3	$8750
9468	48	2.1 GHz	105 MB	4	$9900
9470	52	2 GHz	105 MB	4	$11590
9480	56	1.9 GHz	112.5 MB	4	$12980

Data from: [https://ark.intel.com/content/www/us/en/ark/products/series/232643/intel-xeon-cpu-max-series.html Accessed 11.08.2024].

peak bandwidth may differ when working with HBM [281]. In [279], it is indicated that the maximum bandwidth of this memory in Xeon Max is 1 TB/s (the actual achievable bandwidth is discussed below in Section 4.4). DDR5 memory in Xeon Max models can be used in addition to HBM or absent.

All Xeon Max models (which have far fewer different SKUs than Xeon SPR) use 94xx SKU numbers without suffixes. Although the number of cores in different Xeon Max SKUs ranges from 32 to 56, they all have the same 64 GB of HBM memory and many other similar characteristics, including the same turbo frequency (3.5 GHz) and TDP of 350 W. All Xeon Max SKUs can run on servers with up to two processors. These processors do not support work with Intel Optane [279]. The characteristics of these SKUs are listed in Table 26.

Xeon Max processors support only 1P or 2P configurations. Other data on these processors is provided above in Tables 23 and 24. Given the changes in Xeon Max compared to Xeon SPR described above, only the features of memory modes are discussed below.

Further discussion in this section is based on information from [279]. In total, Xeon Max has 3 basic HBM modes.

HBM-only. The HBM-only mode can be considered as the basic one, in which DDR5 is not used at all. This, of course, assumes working with applications for which 64 GB capacity is sufficient. Such applications can get increased performance due to the higher bandwidth of HBM.

Since each chiplet has its own HBM controller, this mode can also achieve performance gains by using NUMA features.

The application of the ACPI standard to processor frequencies was discussed above in Section 1.2. Also, the ACPI-related static heterogeneous

memory attribute table (containing, in particular, latency and bandwidth, and for Xeon Max, for HBM and DDR), as stated in [279], is used by the Linux kernel to optimize NUMA operation. This allows Linux to pin an application to a specific NUMA node associated with HBM and assign it priority. Support for this NUMA mode is discussed below in the discussion of clustering modes.

Flat Mode (also known as single level memory, `1LM`). This mode can be used if the HBM memory capacity is not enough and DDR5 is used in addition to it. HBM and DDR are configured as main memory in different NUMA address spaces. In this case, not only BIOS (UEFI) configuration is required, but also binding the application to the corresponding NUMA domain using special NUMA software tools.

Cache Mode (also known as dual-level memory, `2LM`). In this mode, unlike Flat mode, HBM is invisible to the operating system and applications, since HBM acts here as a level 4 cache for DDR RAM, not as main memory. In this mode, there is no need to tune applications for it, but the achieved performance may be lower compared to Flat mode.

Clustering modes. Since both HBM and DDR5 memory have their own controllers in each of the four chiplets of the processor, the above 3 basic modes can use more fine-grained NUMA tuning using the `SNC` capabilities described earlier. Such clustering modes are described in the Intel Xeon Max Configuration and Tuning Guide [282]. A detailed discussion of these modes, including graphical illustrations and a discussion of `2P` configurations, is also available in the Lenovo server guide [280]. Information on working with these modes in Linux is also provided in [280, 282].

For each of the three basic HBM operating modes, Xeon Max supports two more clustering modes, using `SNC4` or quadrant, resulting in 6 different modes.

When using clustering with quadrants in HBM-only mode, two different NUMA nodes are formed only in a `2P` server, and in combination with Flat mode, such a server will have 4 NUMA nodes, since each socket has one NUMA node for DDR5 and one for HBM. When working with quadrants in Cache mode, HBM memory is invisible, and there will also be two NUMA nodes for two sockets.

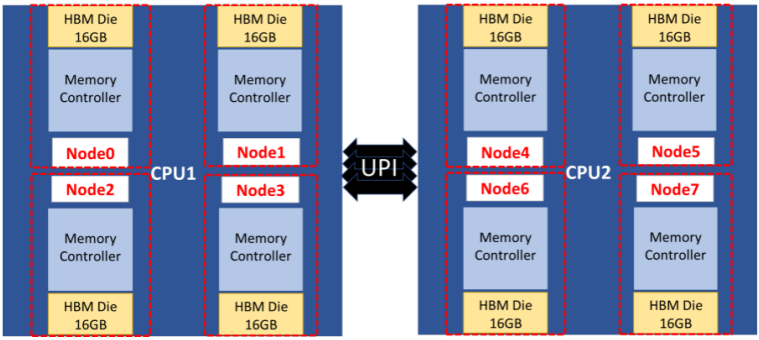


Figure 29. HBM-only mode with SNC4 (figure from [280])

Using SNC4 rather than quadrants offers the potential for higher memory bandwidth and lower latency, as it takes into account the presence of four different controllers and corresponding 16 GB HBM dies for each processor. The highest performance can be achieved by combining SNC4 with HBM-only mode (see Figure 29), which for two sockets gives 8 NUMA nodes. This may require appropriate upgrades in the application and assumes that in a 2P server, 128 GB of HBM is sufficient for the application.

If this capacity is not enough, and DDR5 is also used, then the maximum performance can be achieved by combining SNC4 with Flat mode. Then both HBM and DDR will be divided into four independent NUMA nodes in each Xeon, and in a two-socket server there will be sixteen NUMA nodes.

Finally, it is possible to combine SNC4 with Cache mode. Since HBM acts as a cache and is invisible in this case, a two-socket server will have 8 NUMA nodes. In [280], taking into account the performance data obtained there, general recommendations are formulated for the optimal choice of various NUMA configurations.

Finally, to discuss the various HBM memory modes, it should be noted that in Linux, not only numactl is needed for configuration, but due to memory heterogeneity, daxctl is also needed for renumbering by kernel, as mentioned above.

It should also be noted that although HBM provides clear performance benefits for processors that use it, in addition to increasing their cost, HBM also leads to an increase in the power consumption of processors that integrate HBM. As noted in [283], due to higher power consumption in the 56-core Xeon Max 9480, the frequency (1.9-3.5 GHz) is slightly lower than in the 56-core Xeon 8480+ (2.0-3.8 GHz).

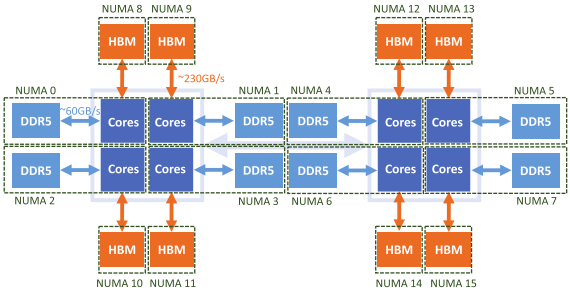


FIGURE 30. Block diagram of the memory of a 2P server with Xeon Max and DDR memory (figure from [284])

Accelerators in Xeon Max. An important advantage for Xeon Max in particular is the AMX ISA extension, since HBM memory can significantly increase the performance of AI tasks. This combination of HBM and AMX tools supporting matrix operations with data in BF16 and FP16 formats, which are relevant for AI, is given special attention, for example, in [14, 284].

It is clear that frequently performed calculations for AI are usually more efficient to perform on GPU. But not for the most complex calculations, especially for AI inference, it may be faster on x86, since there are no delays in data transfer to the GPU. Analysis of where such work on a single server or in a Hadoop cluster of them may be more efficient certainly requires additional research. Below in Section 4 we present the available data on the performance of the Xeon processors considered in the review for AI tasks.

From other accelerators, Xeon Max has DSA (see above in Section 3.1.3); naturally, AiA is also supported.

3.2.1. *Computing systems on Xeon Max*

The NUMA configuration of a 2P server based on Xeon Max in HBM-only mode has already been considered above. The combined use of HBM and DDR memory results in complex NUMA configurations. Figure 30 shows the design of a 2P server with HBM and DDR memory. This allows building NUMA configurations when combining SNC4 with Flat or Cache mode. [280] provides a detailed discussion of such memory configurations and their implementation in Linux.

After the release of Intel Xeon Max with HBM memory, they immediately began to be actively used in scientific research, and supercomputers were created that use these processors in their nodes. The Aurora supercomputer, which appeared much later than planned, contains Intel Data Center Max GPUs and Xeon Max 9470 processors in heterogeneous nodes. The MareNostrum 5 supercomputer, also included in the top ten of the June 2024 TOP500 list, contains homogeneous nodes with Xeon Max 9480 in its GPP partition.

Supercomputers containing only homogeneous nodes with the same Xeon Max 9480 are Crossroads (at the famous Los Alamos National Laboratory) [**278**], which ranks 27th in this TOP500 list, and Camphor 3[34].

Performance data for computing systems on Xeon Max is discussed further in Section 4.

3.3. 5th Generation Xeon Scalable Processors (Xeon EMR)

3.3.1. *Microarchitecture of Xeon EMR Processors and Computing Systems Based on Them*

The Xeon EMR microarchitecture is very close to that described above in Section 3.1 for the Xeon SPR. Accordingly, the description of the Xeon EMR microarchitectural features here is quite limited. The main source of information here is the publication by Intel developers [**285**].

Intel's brief description [**286**] contains very little additional information in this regard, although individual details can be found in other documents, for example, in Intel's guide to monitoring the performance of a not containing cores set of processor units that the company calls "uncore" [**287**]. This set includes CHA (see above in Section 3.1.3 for the Xeon SPR microarchitecture), a power controller unit (PCU), an integrated memory controller (IMC), and others. [**287**] also provides a block diagram of the Xeon EMR, which is shown in Figure 31.

It shows how similar the Xeon EMR and Xeon SPR microarchitectures are (see also Figure 26 for the Xeon SPR above).

[34] https://www.iimc.kyoto-u.ac.jp/en/services/comp/supercomputer/system/specification, accessed 13.05.2025.

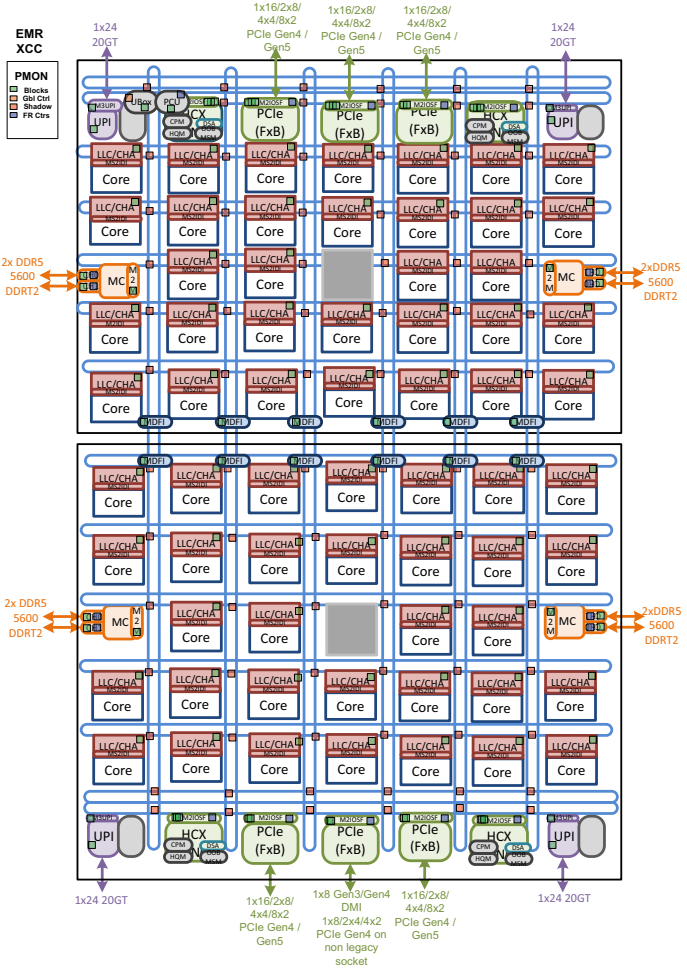


FIGURE 31. Xeon EMR Block Diagram (for XCC class models). Figure from [https://www.intel.com/content/www/us/en/support/articles/000099181/processors/intel-xeon-processors.html Accessed 14.05.2025]

Information about the microarchitecture used by the Xeon EMR cores was provided rather by third-party sources, and scientific publications refer, for example, to Wikipedia [288]. The Xeon EMR cores use the Raptor Cove microarchitecture, a slight improvement over Golden Cove [289]. Although

some publications with data on Raptor Cove have begun to appear [36, 37], but only from the point of view of possible security violations. Therefore, from now on we will immediately use the manufacturer's data for concrete SKUs [290].

In fact, there are 3 SKU groups for Xeon EMR. The Low Core Count (LCC) and Medium Core Count (MCC) models are single-chip. And the Extreme Core Count (XCC) models are multi-chip[35]. These are the ones that Figure 31 refers to.

The major changes for these high-end Xeon EMR models, based on the data in the above references, were the move from four chiplets in the Xeon SPR to two and a large increase (compared to the previous 4th generation of Xeon Scalable processors) in LLC capacity (see Table 23 and Table 24). This was done within the same 10 nm Intel 7 technology, which was optimized, but primarily due to low-level process improvements (see [285]). This also resulted in an increase in the maximum core count from 60 to 64 and improved energy indicators.

Xeon SPR processors contained 4 tiles (chiplets), and Xeon EMR with XCC configuration now contain 2 tiles with 32 cores. According to [289], another design in MCC configuration uses a monolithic die, having from 20+ to 32 cores. And LCC include the models with 20 cores or less [289]. This is the case for specialized EE (edge-enhanced) series processors, Xeon Silver 45xx, which are focused on low power consumption [286] and are not considered in this review. Further consideration of Xeon EMR refers to the XCC configuration.

The increase in LLC capacity, mentioned above as one of the main improvements, applies only to the high-end models with the XCC configuration and a core count of 40 or more. In reality, each XCC tile contains 33 cores, but one is disabled to compensate for the impact of manufacturing defects [289]. The number 33 is due to the fact that the XCC is based on a 7×5 mesh of cores, two of which are replaced by memory controllers [291].

Xeon EMR uses a hemisphere mode (UMA), similar to that used in Xeon SPR. Xeon EMR's lack of support for the previously available NUMA support for various SNC modes (discussed above at the beginning of Section 3) could potentially reduce efficiency of the work with memory. But thanks to

[35] https://www.intel.com/content/www/us/en/support/articles/000099181/processors/intel-xeon-processors.html, accessed 14.05.2025.

the improvement of the manufacturing process, it was even possible to achieve a reduction in the cost of important production stages. And since there are now only two dies in XCC configurations, the number of EMIB interconnect bridges used has been greatly reduced (to three) compared to Xeon SPR [285].

In Xeon EMR has achieved very low operating voltage for LLC [285], and Intel points to power consumption reduction as one of the key areas for Emerald Rapids (see, for example, [289]). Accordingly, performance per watt is improved and TCO is reduced [285, 286, 289, 292].

Unlike Xeon SPR, Xeon EMR processors support only single and dual socket operation. The important improvements of Xeon EMR for processor interconnection over its predecessor is the increase in UPI speed to 20 GT/s from 16 GT/s previously. But this requires some clarification. Xeon Platinum has 4 such UPI channels, Xeon Gold has 3 channels, and Xeon Silver has 2 channels with UPI, both running at 16 GT/s [286].

Another important improvement in Xeon EMR is the support of 8 channels for DDR5: with DDR5-5600 (for 1DPC) and with DDR5-4800 (for 2DPC) for the high-end models. The maximum memory capacity per socket for Xeon EMR is 6 TB, the same as AMD Genoa. For potential expansion of the memory capacity, the support of CXL by these processors may be relevant, which makes it possible to connect more CXL memory in the future. Table 23 and Table 24 also provides other information about the characteristics of Xeon EMR, including clock frequencies, in comparison with the data for EPYC 9004. A huge amount of additional information about the characteristics that have not changed compared to Xeon SPR (for example, support for hyperthreading with 2 threads per core) is not provided here.

Servers based on Xeon EMR are technically similar to servers with Xeon SPR (as noted above, these processors have the same sockets). The main difference is that Xeon EMR does not support servers with 4 or 8 sockets, which was possible for Xeon SPR. Naturally, servers with Xeon EMR are offered by all leading manufacturers. As an example, we will point out the products of the world-famous supercomputer company Lenovo [293]. There are no supercomputers with Xeon EMR in the June 2024 TOP500 list, and in the November 2924 list there are only two systems with heterogeneous nodes, where Nvidia H100 GPUs are used with 32-core Xeon Gold 6548Y; the more powerful of them, the Italian supercomputer PITAGORA, ranks 44th in the TOP500.

There is a lot of data from Intel indicating that Xeon SPR and Xeon EMR are targeted at a wide range of applications. The ability to use accelerators in them that can also be used in a special mode — not with a one-time full payment, but with payment depending on the time of use — provides additional opportunities for narrowly targeting specific SKUs for effective use in a specific area.

Another thing to keep in mind is the separation of capabilities across different processors at the brand level from Platinum to Bronze. [289] notes that Xeon SPR has 52 SKUs, Xeon EMR has 32, and Intel likely has an SKU for every type of workload. It also notes that the EPYC 9004 SKU stack is significantly smaller and easier to understand.

In addition to the Xeon EMR's focus on low TCO, Intel's brief [286] also lists its application areas, from AI and databases to HPC. In [286], the subtitle even states that this is a processor designed for AI. It is clear that the Xeon EMR's success here may be based on the combination of AMX, the use of a higher-performance DDR5 variant, and improved energy efficiency.

Since the possible number of cores in Intel Xeon EMR still lags far behind that available in AMD EPYC 9004, and there is also a lag in the number of memory channels and therefore in its bandwidth, in computationally intensive workloads, including parallelism, the high-end Xeon EMR models are often expected to lag behind EPYC 9004 in performance (see, for example, [289]).

Although some of the Xeon EMR performance data sometimes seems to show a performance penalty relative to the Xeon SPR [289], one should always be careful to clarify the meaning of the data. There is no doubt that the Xeon EMR was Intel's best server processor for traditional dual-socket systems until the introduction of the Xeon 6. This is further illustrated by the cost data (for example, the data in Table 23 shows the reduced price of the top-end Xeon EMR compared to the Xeon SPR). Some rough average cost/performance estimates for different Xeon EMR SKUs are given in [291].

Performance data for systems with Xeon EMR is discussed in the next section.

4. Xeon SPR, Xeon Max, and Xeon EMR Performance Data

We combine the consideration of the performance of several families of Xeon processors into one general section, since they are classified as

belonging to the general conditional 4th generation of x86, have fairly close specifications, and can be replaced within one server, which is ensured by using the same connector (Socket 4677). Such a combination within one document was used, for example, by Fujitsu for information on the performance of its servers with these processors (see, for example, [**166**]).

This is also consistent with the total volume of structured performance data known to us for all these processors (the corresponding servers), comparable to that available for the EPYC 9004. The exception is AI tasks, for which there are many publications, and Intel also provides special optimization guides — for example, for PyTorch [**294**] or for its TensorFlow extension using Python [**295**]. Therefore, AI performance is considered in a separate Section 4.2.

It is also worth pointing out that there is a large amount of performance data on Xeon SPR and Xeon EMR, including performance comparisons, in unstructured form [**296**, **297**], which is not discussed here.

Most of the available and analyzed Xeon performance data relates to the higher-end models with a large number of cores. And a separate Section 4.3 examines the performance of Xeon SPR and Xeon EMR processors in comparison with GPGPU, as well as mid-range Xeon models with similar EPYC 9004 processors.

We also separate the Xeon Max performance data into a separate Section 4.4. This is because the support for two possible memory levels, HBM and DDR, is unique and has caused a surge in scientific publications on the topic. HBM support was discontinued in the subsequent Xeon EMR family, and is absent from the latest Xeon 6 — so the development along the Xeon SPR-Xeon EMR-Xeon 6 line is the main vector of improvement for Intel server processors.

4.1. Xeon SPR and Xeon EMR Performance in Benchmarks and Applications

Our further comparison of the performance of the Xeon SPR with the previous generation Xeon ICL is reasonable limited, since the advantage of the Xeon SPR in this regard is obvious. A very large amount of diverse information for such comparisons is available on the Intel website [**296**]. We illustrate all this with one general Figure 32. It shows the performance data of 2P servers with 56-core Xeon 8480+, Xeon Max 9480 and 40-core Xeon 8380 in applications from a wide range of application areas. Two examples are presented there for the 60-core Xeon 8490H (see [**298**] for more details).

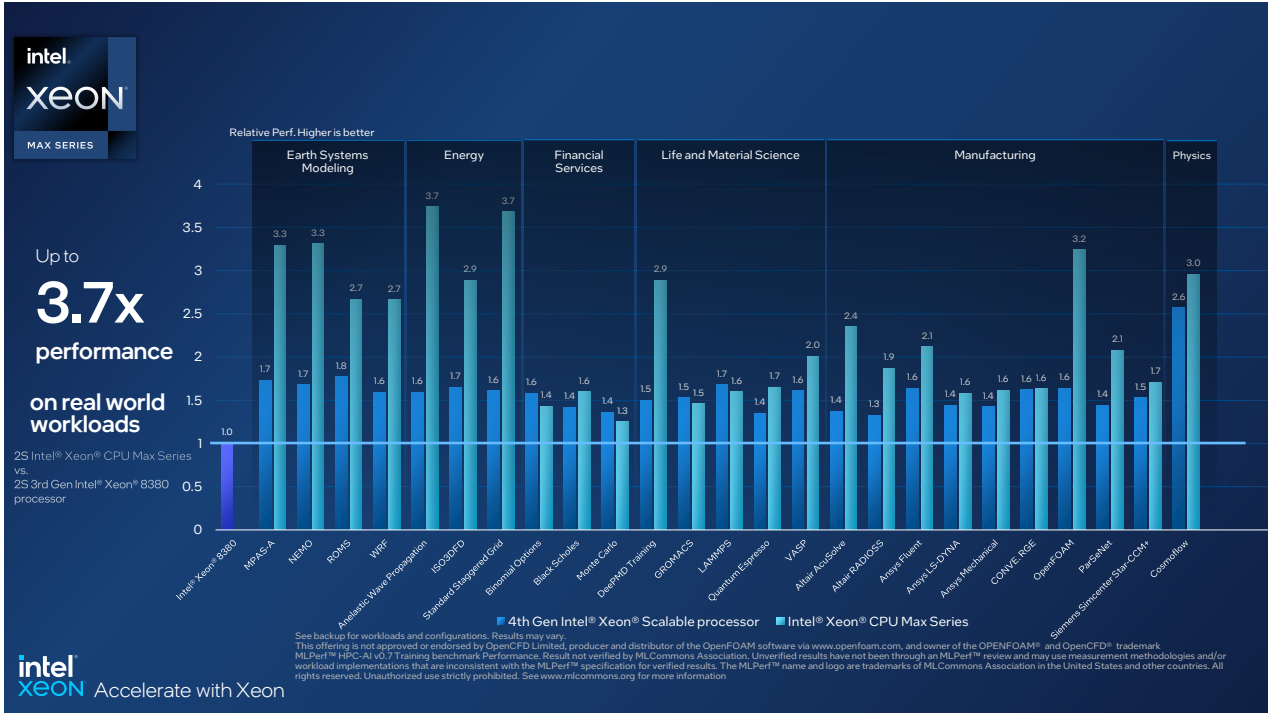


Figure 32. Performance comparison of 2P servers with Xeon 8380, Xeon 8480+ and Xeon Max 9480 (figure from [298])

Performance in various SPEC benchmarks. Table 27 shows the performance of Xeon SPR, Xeon EMR, and EPYC 9004 in SPECcpu 2017 floating point benchmarks. Analysis of the data in this table can provide a number of useful data. It is clear that additional clarifying and explanatory data from scientific publications is often required, but given the high integrity of these indicators and the large-scale attempts to improve them by various server manufacturers, it is useful to draw some conclusions right here.

Similar data for the 3rd generation of Xeon Scalable Processors and EPYC Zen 3 (7003) have already been provided above in Table 1. Comparing these tables, you can see a big improvement in the performance of Xeon SPR compared to the previous generation of Xeon not only due to the increase in the number of cores, but also with the same number of cores (comparing, for example, the 32-core Xeon 8362 and Xeon 8462Y+ models).

Models with more cores usually outperform models with fewer cores, which gives the EPYC 9004 an advantage. With an equal number of processor cores (32), the EPYC 9374F performance is higher than the Xeon SPR 8462Y+ (only in the base variant of fp_speed the EPYC lags slightly behind).

Xeon EMR delivers significant performance gains over Xeon SPR not only due to the higher core counts in the high-end models, but also at the same core counts (for example, compare the 32-core and 60-core Xeon models in Table 27). However, in terms of core counts, the high-end Xeon EMR models still lag far behind the EPYC 9004, which is reflected by the higher EPYC performance numbers in Table 27.

Let's illustrate the performance in SPECcpu by first comparing Xeon EMR and EPYC 9004 processors with the same number of cores. The 64-core Xeon 8592+ processors are slightly behind the EPYC 9534 in performance, while having a slightly higher price. The 48-core Xeon 8558P processors on a 2P server in the fp_rate bencgmark are slightly ahead of the EPYC 9474F in performance, although they are inferior to them in 1P configurations. But the difference in performance is not that great, and you should pay more attention to TDP and cost (and therefore TCO). The price of the Xeon 8558P is slightly lower, but after adding the cost of the chipset, it becomes higher. When comparing 32-core processors, Xeon EMR 8562Y+ and EPYC 9374F, Xeon EMR lags noticeably in fp_rate

benchmarks (possibly more interesting for cloud technologies), and is close in performance to EPYC 9374F in fp_speed benchmark, where the achieved level of parallelization can be low. At the same time, the price of Xeon 8562Y+ is significantly higher.

But the EPYC 9004 has higher core count models, and they outperform the Xeon by a lot more

Individual optimization of separate components of the SPECcpu 2017 fp_speed benchmark (which is reflected in the peak values) has a stronger effect on performance in EPYC than in Xeon. Performance scaling with the number of cores when moving from one processor model to another is not so strong in the fp_speed tests: doubling the number of cores does not provide a performance increase close to twofold (although processors with a small number of cores usually have a slightly higher frequency). But in the fp_rate benchmark, high performance scaling is observed when moving from 1 to 2 processors. This naturally corresponds to the types of these benchmarks themselves.

In both types of SPECcpu 2017 benchmarks, it is not necessarily the case that a small increase in the number of used processor cores when moving to a more expensive processor model of the same architecture leads to an increase in the achieved performance. According to the data in Table 27, the 2P server with the Xeon 8580, which has 4 more cores than the Xeon 8570 at a slightly lower frequency, received lower performance data. Yes, these results depend on the optimization level by the firms performing benchmarks, but in the 1P server, the Xeon 8580 showed higher performance.

It is impossible to say that with the same number of cores in Xeon EMR and EPYC 9004, the performance in these benchmarks is close, based on these data. We will separately conduct a small comparison of mid-range Xeon and EPYC processors, which can also be actively used in HPC [**100**].

Although Table 27 also includes SPECcpu 2017 benchmark data for Xeon Max, their performance will be discussed further in a separate Section 4.4.

SPEC CPU 2017 benchmarks with integer arithmetic are of little interest for this review, especially for HPC. In SPECint_speed 2017, where parallelization has little effect on the result, the Xeon 8592+ is 1% ahead of the EPYC 9684X, and 5% ahead of the EPYC 9654. But in SPECint_rate 2017, the high-end EPYC 9004 models are naturally far ahead (all this applies to the maximum results achieved as of 5.02.2024).

TABLE 27. Comparison of performance of high-end Xeon SPR, Xeon EMR, Xeon Max, and EPYC 9004 models in SPECcpu 2017 floating-point benchmarks

Processor model	Cores per CPU	Frequency, GHz	Price (per 1 CPU)[1]	SPECcpu 2017 fp_speed (base/peak)[2]	SPECcpu 2017 fp_rate (base/peak)[2]
EPYC 9754	128 2×128	2.25–3.1	$11900	316/318[4] 430/448[3]	733/799[4] 1460/1590[3]
EPYC 9654	96 2×96	2.4–3.7	$11805	324/327[6] 449/468[3]	746/784[7] 1480/1570[5]
EPYC 9684X	1×96 2×96	2.55–3.7	$14756	357/358[4] 476/495[3]	820/854[4] 1630/1700[3]
EPYC 9534	1×64 2×64		$8803	298/305[4] 427/442[3]	610/658[4] 1220/1310[3]
EPYC 9474F	1×48 2×48	3.6–4.1	$6780	277/290[4] 405/430[3]	574/575[4] 1150/1150[3,5]
EPYC 9454	1×48 2×48	2.75–3.8	$5225	264/278[9] 388/423[3]	555/557[7] 1100/1110[8]
EPYC 9374F	1×32 2×32	3.85–4.3	$4850	256/272[4] 361/385[3]	481/485[4] 964/968[3]
Xeon 8490H	1×60 2×60	1.9–3.5	$17000	255/—[10] 378/378[10]	508/—[10] 1040/1100[10]
Xeon 8480+	1×56 2×56	2–3.8	$10710	242/242[13] 371/371[10]	465/337[13] 1020/1080[10]
Xeon 8462Y+	2×32	2.8–4.1	—	363/363[10,12]	817/853[10]
Xeon 8592+	1×64 2×64	1.9–3.9	$11600	286/—[14] 428/428[10]	619/646[15] 1260/1300[10]
Xeon 8580	1×60 2×60	2.0–4.0	$10710	286/—[10] 419/419[16]	570/—[10] 1160/1190[16]
Xeon 8570	1×56 2×56	2.1–4.0	$9595	279/—[14] 423/423[10]	544/—[14] 1220/1260[8]
Xeon 8558P	1×48 2×48	2.7–4.0	$6759	275/—[14] 412/412[17]	526/—[14] 1140/1170[17]
Xeon 8568Y+	1×48 2×48	2.3–4.0	$6497	275/—[14] 413/413[17]	526/—[14] 1170/1210[10]
Xeon 8558	1×48 2×48	2.1–4.0	$4650	263/—[14] 412/412[17]	512/—[14] 1090/1120[10]
Xeon 8562Y+	1×32 2×32	2.8–4.1	$5945	261/—[14] 388/388[10]	414/—[14] 908/941[17]
Xeon 8592V	1×64 2×64	2–3.9	$10995	273/—[14] 409/409[16]	542/—[13] 1200/1250[10]
Xeon Max 9480	2×56	1.9–3.5	$12980	348/349[16]	1140/1160[18]
Xeon Max 9470	2×52	2–3.5	$11590	347/349[18]	1100/1120[18]

[1] Data as of 13.05.2024 for AMD — from [49], for Intel — from [https://ark.intel.com/content/www/us/en/ark.html#@PanelLabel595 Accessed 23.02.2025].
[2] Data with maximum peak value represented as of 28.08.2024. Benchmark results submitters and servers where these results were received:
[3] ASUS RS720A-E12-RS12;
[4] ASUS RS520A-E12-RS12U;
[5] xFusion FusionServer 2258 V7;
[6] ASUS RS520A-E12(K14PA-U24);
[7] xFusion FusionServer 1158H V7;
[8] Lenovo ThinkSystem SR675 V3;
[9] Lenovo ThinkSystem SR655 V3;
[10] ASUS ESC4000-E11;
[11] xFusion FusionServer 2288H V7;
[12] ASUS RS720-E11-RS12U(Z13PP-D32);
[13] Supermicro UP SuperServer SYS-521C-NR (X13SEDW-F);
[14] Lenovo ThinkSystem SD530 V3;
[15] IEIT meta brain i24G7;
[16] Nettrix R620 G50;
[17] ASUS RS720-E11-RS12U;
[18] Dell PowerEdge C6620.

Note: To clarify the data on the servers used, the motherboard used in the benchmarks is sometimes indicated in parentheses.

Among the data from various SPEC benchmarks for HPC tasks, SPEChpc 2021 benchmarks are certainly among the most relevant. Of great interest is the work [**299**], which examines in detail the results of SPEChpc 2021 benchmarks (in the tiny variant) for 2P servers with 36-core Xeon ICL 8360Y and 52-core Xeon SPR 8470, and for clusters with these servers (in the small variant of SPEChpc 2021). Although only MPI parallelization options was studied here, this article provides information that goes beyond specific processor models to the level of Xeon ICL and Xeon SPR in general and is of interest for analyzing SPEChpc on newer x86 or other modern processors. In [**299**], individual SPEChpc benchmarks are split into memory-bound and compute-intensive groups, and the performance scaling with the number of MPI processes is investigated in detail under the memory-bandwidth-optimal ccNUMA setup (SNC4 for Xeon SPR). In addition, the power consumption of processors and DRAM, energy consumption, and energy efficiency are examined in detail using EDP (Energy Delay Product) (about EDP, see, e.g., [**300**]).

In [**299**], it was found that the speedup in compute-intensive benchmarkss from SPEChpc when moving from Xeon ICL to Xeon SPR was less than or practically no greater than the increase in the number cores in Xeon SPR compared to Xeon ICL, while memory-bound benchmarks

TABLE 28. Speedup in a server with Xeon SPR compared to Xeon EMR in SPEChpc 2021 (tiny) with MPI parallelization

(a) Computationally intensive benchmarks

	lbm	soma	sweep	sph-exa
Speedup	1.21	1.35	1.39	1.48

(b) Memory-bound benchmarks

	weather	tealeaf	cloverleaf	pot3d	pgmgfv
Speedup	2.03	1.66	1.57	1.63	1.65

received greater speedup (up to two times for the "weather" benchmark) — see Table 28.

Regarding power analyses, in [**299**] defined "hot" and "cold" benchmarks from SPEChpc 2021, which have high and low power dissipation on the processor. The former are compute-intensive benchmarks, where the power consumption is close to the TDP, most of the power is consumed by the compute units and cache memory, and a small part of the power is used in DRAM. Memory-bound benchmarks, on the contrary, are "cold" benchmarks, where high DRAM power is achieved.

DDR5 memory in Xeon SPR has been shown to be much more energy efficient and has a smaller impact on the overall server power consumption than DDR4 in Xeon ICL, although the total memory capacity in Xeon SPR servers was 4x larger. However, the contribution of DRAM to the overall power consumption is small. Xeon ICL and Xeon SPR were found to have high power levels at idle (50% of TDP for Xeon SPR), while Xeon Sandy Bridge, for example, had only 20% of TDP. [**299**] concluded that for compute-intensive benchmarks, Xeon SPR is less efficient, as the increase in power consumption is not offset by the increase in performance.

The results of [**299**] for clusters related with the use of inter-node data exchanges and are not considered here.

Table 29 shows the best "official" SPEChpc 2021 results for the top Xeon SPR, Xeon EMR, and EPYC 9004 models. The bolded values are the maximum base results achieved on traditional 2P servers for each processor model. These values are for server performance, and cluster results are not shown here because they are less representative of the performance of the processors themselves.

Table 29. Performance data for servers with high-end EPYC 9004 and Intel Xeon models in SPEChpc 2021 benchmarks

SKU	Number of CPUs	Tiny base/peak	Small base/peak
EPYC 9654	1 2	6.99/6.99 **13.9**/14.2	0.735/0.735 **1.45**/1.45
EPYC 9684X	2	**16.0**/16.0	**1.55**/1.55
EPYC 9754	1 2	7.32/ **16.4**/	0.823/ **1.59**/
Xeon 8480+	2	**7.98**/8.35	**0.945**/0.949
Xeon 8490H	2 4	**9.00**/ 17.2/17.6	**1.00**/ 1.88/1.89
Xeon 8592+	1 2	4.88/ **10.8**/	0.553/ **1.15**/

Data from [http://spec.org/hpc2021/results/ as of 12.09.2024].

Here we can see that the performance was higher with a higher number of cores used (all cores were used in the processors). But the highest results were obtained on the four-socket server with 60-core Xeon SPR 8490H, although it has slightly fewer cores than the 2P server with 128-core EPYC Bergamo 9754. Compared to Genoa, Bergamo has a smaller L3 cache per core, and its capacity has a significant impact on performance, which is also evident from the comparison of the results for the 96-core EPYC 9684X with 3D V-cache and the EPYC 9654. The success of the four-socket server is probably also due to the fact that performance scales better when using multiple sockets. And among the 2P servers, those with EPYC 9004 have higher performance than those with Xeon.

In a certain sense, the work [**301**] can also be attributed to the SPEChpc 2021 benchmarks, which researched various data on the performance of the CloverLeaf mini-application included in SPEChpc on a 2P server with a Xeon 8480+ and compared it with Xeon ICL.

The next in importance for HPC tasks can be considered SPEComp 2012 and SPECmpi 2007. SPECmpi data for Xeon SPR or Xeon EMR were not available at the time of review writing, and SPEComp data is presented above in Table 7. 2P servers with top Xeon SPR or Xeon EMR models were significantly inferior in performance to top EPYC 9004 models, which is quite natural due to the much smaller number of processor cores than in EPYC models.

We can also note a slight increase in performance in the 2P server with 64-core Xeon 8592+ compared to 56-core Xeon 8480+ or 60-core Xeon 8490H (compared to the latter, the performance increase was about one percent). However, the performance of the 2P server with Xeon 8592+ is close to the performance of the 1P server with the same number of cores (128) with EPYC 9752 Bergamo, which may be due to the reduced L3 cache capacity per core in Bergamo and the increased memory bandwidth when working with the 2P server.

The highest SPEComp indicators were obtained on a server with a Xeon 8490H, but this is the case for an 8-processor server. There was no data available as of the end of 2024 to compare the performance of Xeon and EPYC 9004 on servers with the same number of processors and cores.

From the data of other SPEC benchmarks for Xeon processors compared to EPYC 9004, we will indicate the performance data for Java servers, SPECjbb2015 benchmarks, which were presented above in Table 9. These data show an increase in performance in traditional 2P servers when switching to processors with a higher number of cores in both the Xeon SPR family and the Xeon EMR family. The data about critical_JOPS in the Table 9 for "composite" benchmark on 2P servers with Xeon 8490H compared to Xeon 8480+ do not show an increase in performance, but later the South Korean company KTNF received a higher result on its server, 335807 units.

Performance scaling with the number of Xeon SPR processors in the server up to 8 can be considered quite satisfactory for these benchmarks, except for the "composite" variant, where scaling is poor already when moving from two to four processors.

Comparison of performance with the same number of cores for the 56-core 8480 and 8570, 60-core Xeon 8490H and Xeon 8580 does not show a clear increase when moving from Xeon SPR to Xeon EMR. However, this may simply be due to the small number of results presented so far (the number of optimization attempts) for the Xeon 8570 and 8580. The server with the top model Xeon EMR 8592+ significantly outperformed the server with the top model Xeon SPR 8490H in performance.

But servers with higher-end 4th and 5th Gen Xeon Scalable processors perform significantly worse in these benchmarks than 4th Gen EPYC 9004, and in some cases 2P Xeon servers perform worse than 1P EPYC 9004 servers and 4P Xeon servers perform worse than 2P EPYC 9004 servers, even though the Xeon servers had more cores than the EPYC servers in these comparisons.

In terms of performance per watt, SPECpower_ssj 2008, the Xeon SPR and Xeon EMR outperform the ARM-based Ampere Altra processors (see Table 10 above), and the Xeon EMR shows an improvement over the Xeon SPR. However, the Xeons are significantly slower than the EPYC 9004 in this table.

In addition to the data from this benchmark website, there is interesting SPECpower_ssj 2008 data for 2P Fujitsu servers with 60-core Xeon 8490H and 64-core Xeon 8592+ [**166**], with 32-core Xeon 6438Y+ and Xeon 6538Y+ [**302**], and with 32-core Xeon 6428N [**303**] with various parameters used.

The data above shows the performance advantages of the high-end EPYC 9004 models over the Xeon SPR and EMR. Perhaps the only exception among the SPEC benchmarks (apart from SPECint_speed 2017) where the Xeon SPR or EMR outperformed the EPYC 9004 is the SPECvirt_sc2013 virtualization tests (see Table 11 above). In them, the 2P server with the Xeon EMR 8592+ managed to outperform the 2P server with the EPYC 9654 (the servers with the Xeon SPR lagged behind). These benchmarks, on the one hand, are interesting for mass data centers due to the integration of widespread workloads, but on the other hand, they are less relevant to the performance of the processors themselves, and there are few results for comparison.

Here we can also note the sharp reduction in price of the high-end models of the 5th generation of Xeon Scalable (EMR) processors compared to the 4th generation of Xeon Scalable (SPR) — even the top 64-core Xeon 8593Q model, which requires liquid cooling, costs $12400 (price as of 15.03.2025). But a comparison of performance taking into account prices compared to high-end EPYC 9004 models (see also the discussion above in Section 2.4) does not fundamentally change the advantages of EPYC because of this.

TABLE 30. Memory bandwidth in 2P servers with different Xeon SPR models

Model	Number of cores per CPU	Frequency, GHz	Bandwidth, GB/s
Xeon 8490H	60	1.90	523
Xeon 8480+	56	2.00	518
Xeon 8470Q	52	2.10	492
Xeon 8470N	52	1.70	487
Xeon 8470	52	2.00	511
Xeon 8468V	48	2.40	490
Xeon 8468	48	2.10	485
Xeon 8460Y+	40	2.00	469
Xeon 6458Q	32	3.10	444
Xeon 6454S	32	2.20	445

Data from [**303**].

Memory bandwidth. As a general introduction to the analysis of quantitative memory bandwidth indicators when working with Xeon SPR and Xeon EMR, we can use the data from STREAM benchmarks for 2P Fujitsu servers [**304**]. There, for the STREAM Triad, the dependences of the relative bandwidth values are given for servers with 56-core Xeon 8570 and Xeon 8480+, 32-core Xeon 6548N and Xeon 6430 with disabled SMT on different DIMM types by the number of ranks per memory channel, DRAM column width, DIMM capacity, 1DPC or 2DPC configurations, and the number of DIMMs per socket. In qualitative terms, from the point of view of increasing/decreasing the bandwidth, the influence of most of these parameters is predictable.

The STREAM Triad bandwidth data for a wide range of different Xeon SPR models is presented in Table 30 (data for Fujitsu 2P servers). All servers used in the benchmarks running with 4800 MT/s memory, and the peak memory bandwidth of a single processor for all these models is 307 GB/s.

What we see here is that moving from a lower-end model with fewer cores to a higher-end model shows an increase in bandwidth (when selecting the highest-performing models among those with the same number of cores). In addition, increasing the CPU frequency due to exchanging the processor model while maintaining the same number of cores also usually increases bandwidth.

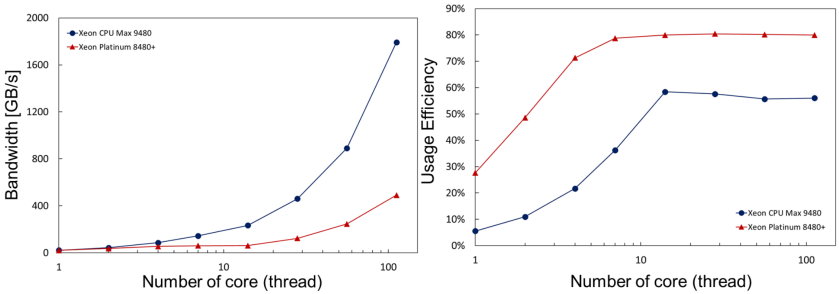


Figure 33. Scaling of the achieved memory bandwidth in 2P servers with Xeon Max 9480 and Xeon 8480+ with the number of cores (figure from [**283**])

As for the Xeon EMR, the Fujitsu server in the 2P configuration demonstrated a significantly higher bandwith (577 GB/s) for the top 64-core Xeon 8592+ model. Such an increase in bandwidth compared to the Xeon SPR naturally leads to an increase in performance for other memory-related benchmarks. Thus, for the two-dimensional FFT in the FFTW benchmark, the 2P server with the Xeon 8592+ provides an average acceleration of 22% (maximum — 68%) compared to the 2P server with the Xeon 8480+ [**305**].

It can be noted that the achieved bandwith of the high-end models of both Xeon SPR and Xeon EMR is significantly inferior to the corresponding data for EPYC 9654, EPYC 9684X and EPYC 9754 (see Section 2.4.2 above), where over 750 GB/s are achieved for 2P configuration. But if we calculate the bandwidth per core from these numbers, then the 60-core Xeon 8490H and 64-core Xeon 8592+ have higher bandwidth than the 96-core EPYC processors.

The dependence of memory bandwidth in a 2P server with Xeon 8480+ in the STREAM benchmark on the number of used cores (on the abscissa axis, in logarithmic form) is presented in Figure 33.

It shows that bandwidth scaling takes place to a certain extent up to the maximum number of available cores. Although the efficiency of using additional cores decreases after a certain threshold.

Of particular interest are the memory bandwidth data for 4- and 8-processor Fujitsu servers with Xeon SPR. For a 4-processor server with

60-core Xeon 8490H in STREAM Triad, 805 GB/s was achieved versus 1600 GB/s for an 8-processor server. For a 4-processor server with 40-core Xeon 8460H, the achieved bandwidth was 817 GB/s versus 1593 GB/s for an 8-processor server. These data are presented in [**63**] and [**306**], and demonstrate expectedly good scaling with an increase in the number of processors used.

Performance in dense matrix multiplication and HPL benchmark. Unfortunately, the author is not aware of the performance data of 2P servers with top models of Xeon SPR and Xeon EMR (not their individual cores) using DGEMM tools from MKL for dense matrix multiplication in FP64 format. However, for the 48-core Xeon 8468, which has a peak performance of about 3.2 TFLOPS, there are data for the Japanese supercomputer Pegasus, which uses these processors in nodes together with Nvidia H100 GPUs [**307**]. There, using oneMKL, a performance of about 3 TFLOPS (93% of the peak) was achieved, while enabling the turbo mode there did not provide significant acceleration.

For FP32, there are data on the corresponding execution times on a 2P server with a Xeon 8480+ using MKL tools [**4**].

It is also necessary to note here the works using AMX, which relate not only to AI tasks, but also to mixed-precision calculations in general (see, for example, [**308**]). AMX is also used in oneMKL. Intel in [**309**] provides data on the performance of a 2P server with 56-core Xeon 8480+ when multiplying square matrices of different sizes using AMX — in FP32 and BF16 formats. This information is presented in Figure 34.

There is also some interesting data comparing the performance of using AMX for GEMM and GEMV in BF16/FP16 formats in a 2P server with 40-core Xeon 8460H compared to Nvidia P100 and A100 GPUs, with the two Xeon processors more often outperforming the P100 in GEMM across different matrix sizes [**310**].

For these formats, the performance of a 2P server with a Xeon 8480+ using other GEMM software implementations is presented in [**311**], where it is compared, in particular, with data for a 2P server with AWS Graviton 3 ARM processors containing 64 Neoverse V1 cores. Naturally, the performance achieved by the Xeon 8480+ turned out to be much higher. The corresponding data are discussed further in Section 4.2 on Xeon performance in AI tasks.

These works on half-precision GEMM performance we referenced above focused on AI tasks and, in particular, large language models.

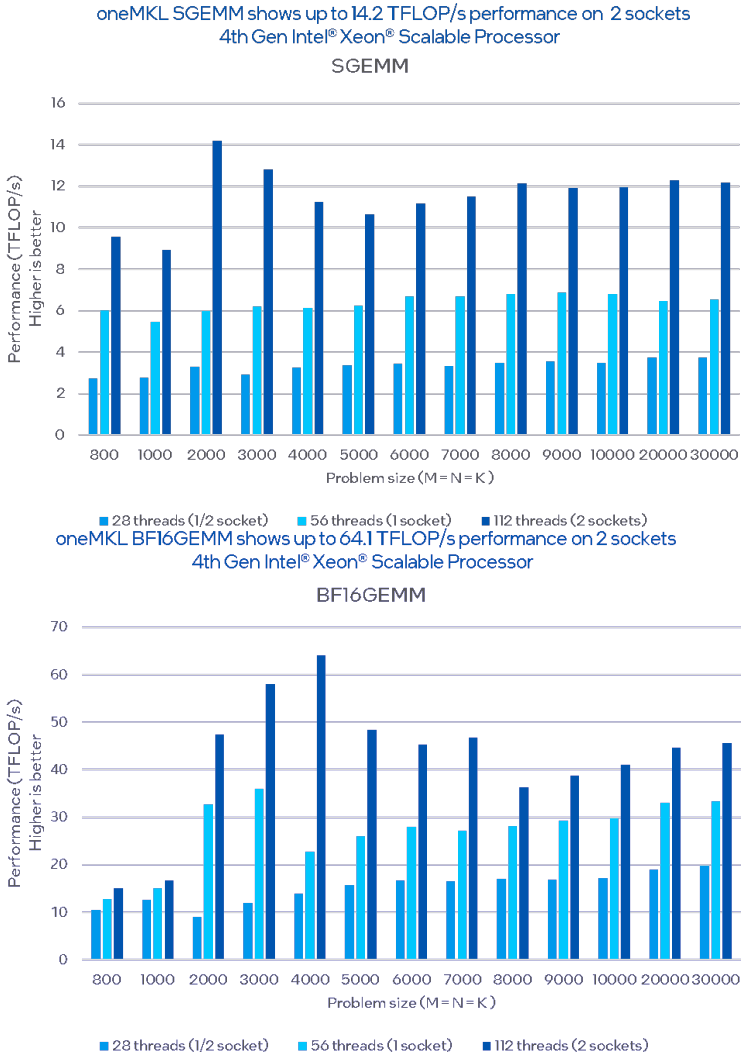


FIGURE 34. Performance of a 2P server with Xeon 8480+ when multiplying matrices in FP32 and BF16 formats (figure from [**309**])

Performance data for Xeon SPR and Xeon EMR in the HPL benchmark was provided by the manufacturers of servers with these processors.

TABLE 31. Performance of 2P servers with Xeon SPR in the HPL benchmark

Model	Number of cores per CPU	Frequency, GHz	Performance, GFLOPS, Data from	
			[303]	[166]
Xeon 8490H	60	1.9	7584	7386
Xeon 8480+	56	2.0	7293	7388
Xeon 8470Q	52	2.1	7051	
Xeon 8470N	52	1.70	6264	6105
Xeon 8470	52	2.00	7051	6930
Xeon 8468V	48	2.4	7022	6321
Xeon 8468	48	2.1	6698	6544
Xeon 8458P	44	2.7		6162
Xeon 8460Y+	40	2.00	5538	5421
Xeon 8462Y+	32	2.80		5522
Xeon 6548Q	32	3.10	6160	
Xeon 6454S	32	2.20	4667	4418
Xeon 6428N	32	1.80	4025	3826
Xeon Max 9480	56	1.90	6781	

Table 31 shows the performance data for Fujitsu PRIMERGY servers with Xeon SPR and Xeon EMR in HPL — for the RX2540 M7 [**166**] and for the CX2550 M7 [**303**].

The data in Table 31 provide some estimates of how performance may change with an increase in the number of cores or clock frequencies. It is clear that the data presented in this table from Fujitsu may not be the maximum achievable values, which depend, in particular, on the size of the matrices used. For the Xeon 8592+, [**303**] indicates a performance of 8542 GFLOPS. These performance values for the Xeon 8480+, Xeon 8490H, and Xeon 8592+ are higher than those for the EPYC 9654, shown in Table 15. However, the maximum performance given by AMD in [**121**] for a 2P server with the EPYC 9654, 8856 GFLOPS, is significantly higher than that of the Xeon SPR, and slightly higher than that of the Xeon 8592+. Similar data in [**121**] for EPYC 9684X and EPYC 9754 (they are given in Section 2.4.3) are also higher than those of these Intel processors.

Table 32. HPL performance on a server with Xeon 8480+

Matrix size N	140000	180000	200000	220000
Performance, GFLOPS	4414.2	4683.8	4764.9	4819.0

Table 33. HPL performance scaling with number of Xeon SPR processors

Model	Number of cores	Frequency	Performance, GFLOPS			
			4 processors		8 processors	
			HPL	Peak	HPL	Peak
Xeon 8490H	60	1.90 GHz	14505	14505	25061	29184
Xeon 8468H	56	2.10 GHz	12656	12902	23274	25805
Xeon 8460H	52	2.20 GHz	11872	11264	21697	22528

This is primarily due to the larger number of cores in these EPYC models than in Xeon SPR and Xeon EMR. A certain lag of EPYC compared to Xeon in the number of floating-point operations per clock in EPYC cores, in contrast to matrix multiplication, is partly leveled out here, since in HPL the total calculation time is affected not only by matrix multiplication.

In [**313**] for a server with 56-core Xeon 8480+ processors, the Japanese company HPC Systems provides data on performance in HPL depending on the matrix size N. The corresponding data are presented in Table 32.

These results were obtained using the Intel classic compiler from oneAPI Base & HPC Toolkit 2022.2.0, as well as the corresponding version of oneMKL.

In addition, there are data on the performance of Xeon SPR in HPL on 4- and 8-processor servers from Fujitsu [**63**, **306**], which are presented in Table 33.

These data indicate good scalability of HPL performance with the number of processors in the server and its proximity to the peak value. It should be noted here that turbo mode was enabled in the tests, and peak performance is calculated at base clock frequencies.

While traditional 2P servers with high-end Xeon SPR and Xeon EMR models perform worse in HPL than servers with high-end EPYC 9004 models, 4- and 8-socket Xeon SPR servers outperform them because EPYC 9004 does not support such multi-socket configurations.

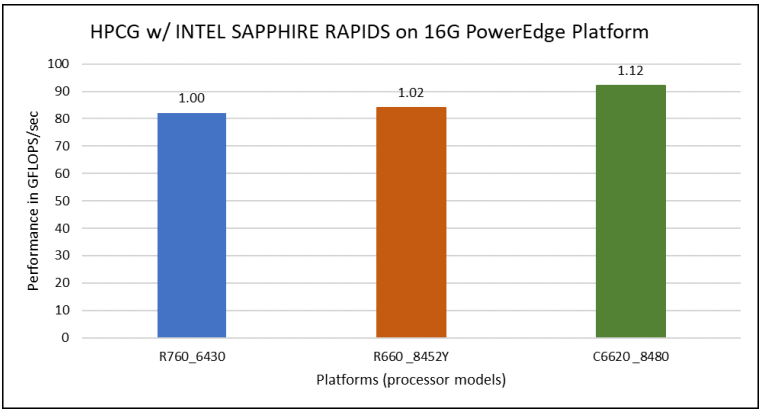


FIGURE 35. HPCG performance in 2P servers with Xeon SPR (figure from [314])

HPCG benchmarks. In contrast to the computationally intensive HPL performance data discussed above, HPCG was a supercomputer-oriented, memory-bound benchmark from the outset (see, e.g., [313]), and was considered to better match real-world applications. HPCG's aspirations to the reference level were realized in the form of a separate list of supercomputers by HPCG performance on the TOP500 website, with the latest available at the time of review writing November 2024 version[36] showing the Fugaku supercomputer with homogeneous ARM A64FX nodes continuing to lead. This demonstrates that GPU-less supercomputers have not lost their relevance. Accordingly, it seems natural to consider Xeon's performance on the HPCG benchmark immediately after the HPL performance data.

HPCG performance data (in GFLOPS) for different Dell PowerEdge servers with Xeon SPR (with 32-core Xeon 6430 and 8452Y models and 56-core Xeon 8480+ model) is presented in Figure 35.

This data can be used in terms of the relative performance of different Xeon SPR models, since the exact problem size for this figure is not given in [314].

However, HPCG performance data is available for a wider range of processors, including Xeon EMR and EPYC 9004 [202, 289], including those conducted as the HPCG 3.1 PTS benchmark [202], which are presented in Table 34.

[36] https://top500.org/lists/hpcg/list/2024/11/, accessed 7.05.2025.

Table 34. Performance of servers with Xeon SPR, Xeon EMR, and EPYC 9004 processors in the HPCG benchmark

Model	Number of cores per CPU	Server Performance, GFLOPS	
		1P-configuration	2P-configuration
Xeon 8592+	64	35.42	70.95
Xeon 8490H	60	31.24	60.42
Xeon 8380+		20.65	40.31
EPYC 9684X	96	23.78	44.11
EPYC 9654	96	32.13	43.97
EPYC 9684X	96		With Power determinism: 73.26
EPYC 9654	96		With Power determinism: 50.86

Note: runtime 60 sec., submesh dimension 144.

Naturally, including due to the higher memory bandwidth, the Xeon SPR (model 8490H) is significantly ahead of the Xeon ICL (model 8380) according to this table, and the Xeon EMR (model 8592+) is faster than the Xeon SPR. When working with the default BIOS settings in the EPYC 9004, the high-end Xeon SPR and Xeon EMR models significantly outperformed the EPYC 9004 in HPCG performance. When switching on Power determinism in the BIOS for the EPYC 9004, which enables higher clock frequencies by increasing the TDP (see Section 2.2 above), the performance of the EPYC 9004 increases significantly. However, both Xeon EMR and Xeon SPR are still significantly faster than the EPYC 9654 processor, which is often used in supercomputers, and only the 2P server with EPYC 9684X slightly outperformed the 2P server with Xeon 8592+.

Here the question arises of the level of scaling of HPCG performance with the number of cores in multi-core processors (for 96-core EPYC 9004 in Table 35 this is more important than for Xeon) when using 144 as the local submesh size for all three coordinates in [**202**] (the resulting submesh is assigned to each MPI process, see [**315**]). In [**314**] higher performance was obtained for Xeon 8480+ than in [**202**] for Xeon 8592+, and there a submesh size of at least 192 was used. And according to AMD [**121**], the achieved performance of high-end EPYC 9004 models in HPCG with a submesh size of 192 is a couple of times higher than in these PTS benchmarks.

NAS Parallel Benchmarks (NPB). The performance dependence graphs in these benchmarks for 32-core Xeon SPR models are shown above in Figure 15 and Figure 16.

TABLE 35. Performance of compared Xeon and EPYC generations in NPB 3.4

Model	Number of CPUs	Performance, MOP/s	
		BT.C	SP.C
Xeon 8490H	1×60	180757	72703
	2×60	336632	152641
Xeon 8592+	1×64	236491	111687
	2×64	437264	242574
EPYC 9554	1× 64	233622	119472
	2×64	452577	242455
EPYC 9654	1×96	261865	125374
	2×95	509783	263543
		551536(*)	272752*
EPYC 9684X	1×96	312136	208538
	2×96	625625	353106
		708203(*)	390558*

(*) are marked EPYC results with Power determinism (others are for default mode).

Here, we limit ourselves to the analysis of the data [**202**], which provides performance and energy efficiency data in the corresponding NPB PTS benchmarks for different generations of Xeon Scalable processors, including Xeon ICL, Xeon SPR, and Xeon EMR, as well as EPYC 9004 Genoa and Bergamo.

Table 35 provides performance data for 1P and 2P servers with Xeon 8490H, Xeon 8592+, as well as EPYC 9654, EPYC 9684X, and EPYC 9554. It provides data only for EPYC Genoa, since they are typically used in HPC, rather than Bergamo with less computationally efficient Zen 4c cores. But the 128-core EPYC 9754 with Zen 4c, like the high-end Genoa models, outperformed the Xeon SPR and Xeon EMR in NPB performance [**202**].

The table includes data from [**202**], which did not show gross performance scaling violations with core count (this may also be due to the memory-bound nature of the NPB benchmarks). According to the data from [**202**], in all benchmarks, EPYC Genoa outperformed Xeon SPR and Xeon EMR, both in 1P and 2P configurations, and sometimes this was the case with the same number of cores in Xeon and EPYC.

This was the case both in default determinism and in Power mode, which usually provided a significant performance increase for Genoa. In terms of performance per Watt, these Xeon SPR and Xeon EMR models also lagged behind EPYC 9004 [**202**].

A more wide set of performance data for different processor models in NPB PTS benchmarks is available[37]. Some data for the NAS FT.C benchmark (with 3D FFT) is provided above in Table 17.

In this benchmark, Xeon SPR and Xeon Max processors lag behind the high-end EPYC 9004 models (this is also the case for Xeon EMR), and sometimes behind the EPYC Milan. But in NPB PTS benchmarks, you also need to pay attention to the correspondence of the used classes (task sizes) to the scaling of performance with the number of cores. In the LU.C test, where the lack of performance scaling with core count was not outwardly apparent, 1P and 2P servers with EPYC 9684X, EPYC 9654, and EPYC 9554 were faster than similar servers with Xeon SPR and Xeon EMR, and 1P and 2P servers with 32-core EPYC 9374F outperformed servers with 60-core EPYC 8490H (see also the discussion of EPYC 9004 performance in NPB PTS benchmarks in Section 2.4.5 above).

Graph500 benchmarks. The benhcmarks discussed above were primarily focused on HPC tasks and are relevant for supercomputers as well. Among the benchmarks that are potentially interesting for modern supercomputers, we can note the performance analysis for intensive data processing, which is carried out in the Graph500 benchmark for the task of working with graphs. The corresponding data for the Graph500 3.0 PTS benchmarks on 1P and 2P servers with Xeon 8592+, Xeon 8490H and Xeon ICL 8380 are presented in **[202]**. It is clear there that Xeon SPR significantly outperforms Xeon ICL in performance, and Xeon EMR, accordingly, outperforms Xeon SPR. However, all Xeon processors significantly lag behind the high-end EPYC 9004 models in performance.

Benchmarks for cloud technologies. Various benchmarks, conditionally integrated in this review into the group of those actual for cloud technologies (as was done earlier and when considering the performance of EPYC 9004), naturally include online analytical processing (OLAP). Features for OLAP using chiplets of modern multi-core processors (primarily NUMA) are considered in **[316]**. In particular, data on TPC-H performance were obtained there for a 2P server with 56-core Xeon 8480+, as well as a 2P server with 64-core EPYC Milan 7713 and a single-processor server with a 64-core ARM processor AWS Graviton 3.

[37] https://openbenchmarking.org/test/pts/npb, accessed 12.03.2025.

Data on the acceleration in TPC-H, obtained on a 2P server with 64-core Xeon 8592+ compared to a similar server with 56-core Xeon 8480+, are presented in [**305**]. For the OLTP-2 test, which is a certain developed by Fujitsu analogue of the TPC-E standard, in [**303**] estimates of the results are presented for servers of this company with different models of Xeon SPR (from 8- to 32-core) and Xeon EMR (from 8- to 24-core).

Other actual benchmarks classified here in this group are the standard SAP benchmarks[38]. In [**63**], performance data was obtained on an 8-processor Fujitsu server with 60-core Xeon 8490H for all three phases of the SAP Business Warehouse (SAP BW) benchmark for the SAP HANA database. And for a 2P Fujitsu server with Xeon 8490H, performance data was obtained in [**166**] in the two-level standard SAP SD benchmark, and an acceleration of 1.6 times was achieved compared to a 2P server with 40-core Xeon ICL 8380 (which is close to the ratio of the number of cores in these servers). [**305**] provides performance data on SAP BW for HANA obtained on a 2P server with 64-core Xeon 8592+.

Database benchmarks may also be actual for cloud technology. Here we point to the PostgreSQL PTS benchmark data in [**202**] on performance with Xeon 8592+ compared to 56-core Xeon 8490P+ and high-end EPYC 9004 models. In these benchmarks, 2P servers with the specified processors provide fewer transactions/s than 1P configurations. As for 1P servers, compared to a server with Xeon 8490H, the Xeon 8592+ processor provides a 6.6% acceleration, and the faster 96-core EPYC 9654 — a 10.8% acceleration.

Performance of continuum mechanics tasks. The relative performance of the 40-core Xeon ICL 8380 and 56-core Xeon SPR 8480+ in the OpenFOAM CFD software benchmarks, the ANSYS Fluent, LS_DYNA, and Mechanical CFD applications, and the WRF weather forecast application is shown above in Figure 32, and for the Xeon SPR it shows an increase of approximately one and a half times, which is approximately how many times the number of cores has also increased.

Since the performance of CFD applications is often limited by sparse matrix-vector multiplications, in [**317**] the possibilities of converting such multiplications (SpMV) to sparse matrix products (SpMM) for CFD tasks were investigated on a 2P server with 56-core Xeon 8480+ to achieve higher performance.

[38] https://www.sap.com/dmc/exp/2018-benchmark-directory/#/q2c, accessed 16.03.2025.

Table 36. Calculation time on 1P and 2P servers with different processor models in the OpenFOAM drivaerFastback PTS benchmark with medium mesh size

Model	Number of cores	Configuration 1P	Configuration 2P
Xeon 8592+	64	302.1	137.0
Xeon 8490H	60	420.6	196.5
Xeon Max 9480	56	410.3	186.3
EPYC 9654	96	325.4	124.3
EPYC 9684X	96	178.2	93.6

The LULESH (mini-application for solving of Sedov explosion problems) PTS benchmark, as noted above in Section 2.4.7, does not scale well with the EPYC 9004 core count (from a lower core count model to a higher core count model — for example, a 2P server with 64-core EPYC 9554 was faster than with 96-core EPYC 9654 [**202**]). The best performance in this benchmark was demonstrated by the 2P server with 64-core Xeon 8592+, which was 1.55x faster than the 2P server with EPYC 9654r, and 1.61x faster than with 60-core Xeon 8490H. And in terms of performance per Watt, servers with Xeon 8592+ outperformed servers with high-end EPYC 9004 models [**202**].

In the OpenFOAM 10 drivaerFastback PTS benchmark with a small mesh size [**318**], the 2P server with the Xeon 8592+ outperformed all servers with the EPYC 9004, except for the 2P server with the EPYC 9684X, lagging behind the latter by only 4%. But with the medium mesh size, the more frequently used EPYC 9654 also outperformed the Xeon 8592+, by 10%. Table 36 presents the data [**318**] on the time of the corresponding calculations for the medium mesh size.

The Xeon 8592+ significantly outperforms the Xeon 8490H here, which is likely largely due to the faster memory used. Performance also increases due to the use of HBM memory (in the Xeon Max 9480) or due to the use of 3D V-cache (in the EPYC 9684X), but the 96-core EPYC 9004 models outperform the Xeon regardless of the presence of such improvements in the memory hierarchy. These tables show the expected good scaling of performance when moving from a 1P to a 2P configuration.

Report [**320**] demonstrated the effect of enabling OPM mode in BIOS for the Xeon 8592+ for the OpenFOAM motorbyke PTS benchmark. The compute time was about 1 second longer in this mode, but the Xeon's average 2P power consumption was almost 80 W lower, and the peak power

consumption was 117 W lower. For the more computationally demanding drivaerFastback model, the performance in OPM mode was comparable to the baseline, but with OPM the average 2P power consumption was 33 W lower, and the peak power consumption was 37 W lower. However, both of these benchmarks were run with a small mesh size.

In the WRF 4.2.2 PTS benchmark with "conus 2.5km"as the input data in [318], the 2P server with Xeon 8592+, as with the OpenFOAM drivaerFastback benchmark with a small mesh size discussed above, outperformed the 2P servers with Xeon 8490H (by 19%) and with EPYC 9654 (by 24%), i.e. the latter also fell behind the 2P server with Xeon 8490H. But the leader in performance here was the 2P server with EPYC 9684X, which outperformed the server with Xeon 8592+ by 26%. In addition, 1P servers with EPYC 9684X and EPYC 9654 were faster than 1P servers with Xeon 8592+ (although EPYC 9654 was only 6% faster). EPYC 9654 was 18% faster than Xeon 8490H.

In [289], in a similar WRF benchmark with "conus 2.5km" (there are also data for a 2P server with 56-core Xeon 8480+), the 2P server with Xeon 8592+ outperformed the 2P server with EPYC 9654, but lagged behind the 2P server on the EPYC 9554 with same number of cores (64 per CPU). All these benchmarks require clarification of the situation, primarily by considering the dependence of performance on the number of cores used in the calculation.

Based on the above data from the LULESH, OpenFOAM and WRF PTS benchmarks, we can see a significant increase in the performance of the Xeon 8592+ compared to the Xeon 8490H. We can say that the Xeon 8592+ is competitive in performance with the high-end EPYC 9004 models. The Xeon 8592+ sometimes outperformed the EPYC 9654 in performance, but this was true for tasks of rather small dimensions. It is important to clarify the available data in terms of determining the dependence of performance on the number of cores used in the servers.

Scaling of OpenFOAM and WRF performance with the number of cluster nodes with Xeon 8480+ and Infiniband NDR 200 is discussed in [320].

Performance for computational chemistry tasks. Next, we will consider information about the performance of Xeon SPR and Xeon EMR in molecular dynamics, quantum chemistry, and the miniBUDE mini-application for molecular docking, which is conditionally classified here as computational chemistry. The largest amount of such data traditionally relates to molecular dynamics, including due to the often high performance scaling indicators with an increase in the number of cores used.

Table 37. Comparison of GROMACS performance (ns/day) in servers with Xeon SPR, Xeon EMR with servers on other processors

Model	Number of cores	Configuration 1P	Configuration 2P
Xeon 8592+	64	10.5	18.3
Xeon 8490H	60	8.6	15.2
Xeon 8380	40	4.8	9.0
EPYC 9654X	96	11.5	18.8
EPYC 9654	96	11.4	19.0
EPYC 9554	64	9.7	16.9

For the GROMACS molecular dynamics application, which is often considered the most well-parallelized, the GROMACS 2023 PTS benchmark performance data for 1P and 2P server configurations with 64-core Xeon 8592+, 60-core Xeon 8490H, and 40-core Xeon 8380+ were obtained in [202] for the water_GMX50_bare input data (computing the complexes of large number of water molecules) using MPI parallelization. The corresponding performance (number of nanoseconds of the system's dynamics simulated in one day of computing) is presented in Table 37, where it is also compared with the high-end EPYC 9004 models.

The data in this table shows a large increase in performance when moving from the high-end Xeon ICL models to the high-end Xeon SPR models and then to the Xeon EMR. When moving to the Xeon SPR, the achieved performance increase exceeded the increase in the number of cores. A similar thing happens when moving from the Xeon SPR to the Xeon EMR.

Both the Xeon and EPYC 9004 models presented in the table show good performance scaling when moving from 1P to 2P server configurations. Servers with high-end EPYC 9004 models outperformed servers with high-end Xeon models due to their higher core counts. Performance in this benchmark increased with further increases in the number of cores in the model. Even higher performance than presented in the table was achieved in [202] on a server with 128-core EPYC 9754. In addition, the performance of servers with EPYC 9004 increases slightly when Power determinism is enabled.

When comparing servers with the same number of cores — with 64-core Xeon 8592+ processors and with EPYC 9554 — the performance of Xeon in this benchmark is about ten percent higher. However, no data is provided for calculating with EPYC 9554 with Power determinism in [202].

According to [**202**], in terms of energy efficiency, Xeon processors were inferior to the high end EPYC 9004 models.

Other performance data for Xeon 8592+ and Xeon 8490H servers, presented in Table 37, were obtained in [**318**] for the GROMACS 2024 PTS benchmark (with GROMACS 1.9) for the same input data with the same MPI parallelization, and also compared to the EPYC 9004. These results are close to those presented in Table 37, and are not considered here.

A broader coverage of performance data in the same GROMACS PTS benchmarks variants for different Xeon processors, and for other processors in general, is available[39]. These data show that 1P servers with Xeon 8592+ and Xeon 8490H outperform 1P servers with modern ARM processors — with 192-core AmpereOne and with 96-core Neoverse-V2 (probably AWS Graviton 4).

As for the performance data of Xeon SPR and Xeon EMR in another widely used molecular dynamics application, NAMD, the performance of a 2P server with Xeon 8490H in the Her1-Her1 protein homodimer benchmark containing 456 thousand atoms is presented in [**321**].

According to AMD [**195**], 2P servers with 128-core Bergamo processors (EPYC 9754) significantly outperformed 2P servers with 56-core Xeon 8480+ in several benchmarks with different initial data, which may be natural given good scalability with the number of cores. And AMD reported significantly higher NAMD performance in a 2P server with EPYC 9654 compared to a server on Xeon 8480+ in [**196**] without specifying specific initial test data. Data on the performance of servers with some Xeon SPR models in NAMD PTS benchmarks is available[40].

Performance data for Xeon SPR and Xeon EMR in LAMMPS PTS benchmarks version 1.4, for another well-known molecular dynamics application, with calculations of the rhodopsin protein and a molecular system of 20 thousand atoms, is available[41]. Although according to these results, Xeon EMR and Xeon SPR lagged behind the high-end EPYC 9004 models in performance, this requires additional, more detailed analysis of the achieved performance, including a study of the dependence on the number of cores used in the calculations.

[39] https://openbenchmarking.org/test/pts/gromacs, accessed 23.03.2025.

[40] https://openbenchmarking.org/test/pts/namd, accessed 24.03.2025.

[41] https://openbenchmarking.org/test/pts/lammps, accessed 24.03.2025.

With the NWChem 7.0.2 PTS benchmark, for the NWChem quantum chemistry software suite with highly advanced parallelization capabilities, performance data for 1P and 2P servers with Xeon EMR and Xeon SPR, including Xeon 8592+ and Xeon 8490H, were obtained in [**202**]. These data relate to the DFT calculation of the C240 molecule, which was discussed above in Section 2.4.8 in terms of EPYC 9004 performance. Servers with Xeon 8592+ outperform here servers with Xeon 8490H by about ten percent. Servers with EPYC 9654 also significantly outperform systems with Xeon 8592+ here. But as noted in Section 2.4.8, this benchmark with the processors considered in this review is characterized by poor scalability with the number of cores involved in the benchmark, and, in particular, when moving from 1P to 2P configurations. Here, for example, a 1P server with a 64-core EPYC 9554 showed higher performance than systems with a 96-core EPYC 9654.

A larger set of performance data for different processors in this benchmark is available[42]. According to this data, the 1P server with a 72-core ARMv8 Neoverse-V2 (probably Nvidia Grace) required less calculation time in this benchmark than the 64-core EPYC 9554.

Another quantum chemistry application for which performance data is available with a 56-core Xeon 8480+ is CP2K. In [**320**], CP2K performance scales well in DFT and RI-MP2 methods for a 16-node cluster with Infiniband NDR 200. And for the Quantum ESPRESSO application, in [**195**], AMD reported that a 2P server with a Xeon 8480+ lagged behind a 2P server with 128-core EPYC 9754 by about 1.4 times. Strictly speaking, this calculation belongs to quantum molecular dynamics by the Car-Parrinello method, but the computation time here is limited by the calculation of forces by the quantum chemical DFT method.

A mini-application of molecular docking, miniBUDE, is often used in benchmarks on modern computing systems. In the miniBUDE PTS benchmark with OpenMP parallelization, performance data were obtained in [**202**], in particular, for 1P and 2P systems with Xeon 8592+ and Xeon 8490H. The performance of this benchmark scales well when moving to high-end models with a larger number of cores and from 1P to 2P configurations. The performance reported below applies to 2P servers. The

[42] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.

64-core Xeon 8592+ turned out to be faster than the 60-core Xeon 8490H by a factor of 1.38 — much more than the increase in the number of cores. However, the Xeon 8592+ lagged behind the EPYC 9654 in performance with Power determinism by 1.88 times, while the number of cores in the Xeon is one and a half times less than in the EPYC 9654.

In terms of energy efficiency in the miniBUDE benchmark, computing systems with EPYC 9004 in [202] also far outperformed servers with Xeon EMR and Xeon SPR.

The performance of Xeon SPR and Xeon EMR in AI tasks will be discussed below. Therefore, it seems reasonable to give a brief conclusion here on the performance data in widely used benchmarks, including application benchmarks. It can be said that Xeon EMR (most of the data was related to Xeon 8592+) significantly outperforms the high-end Xeon SPR models in performance and also has cost advantages.

When comparing the performance of Xeon EMR and EPYC 9004, you should, of course, focus on a concrete application area. But in general, it can be said very roughly that in terms of performance (primarily in the HPC area), the high-end Xeon EMR and, accordingly, Xeon SPR models usually lag behind the high-end EPYC 9004 models with a higher number of cores. When comparing servers with a similar or identical number of cores, Xeon EMR not infrequently achieved higher performance, but at the same time lagged behind the EPYC 9004 in terms of energy efficiency and cost.

4.2. Xeon SPR and Xeon EMR Performance in AI Tasks

Performance data for AI tasks of computing systems (from servers to a small cluster) based on Xeon SPR, compared with similar results for 3rd generation Xeon Scalable processors and for Zen 3 processors, is presented by Intel in [322].

Information on the acceleration of Xeon SPR by leveraging AMX capabilities in AI workloads is available in [118]. The previous section provides data on the performance of Xeon SPR in AI-relevant GEMM operations in the BF16 format supported by AMX. At the end of this section, GEMM performance is also illustrated for FP32 and BF16 formats compared to ARM processors.

Among the publications aimed at increasing AI performance using AMX, we note the works [310, 323–326]. It is clear that the achieved acceleration value may vary. For example, in [325], modifications to the software using AMX provided an acceleration of up to 1.5 times. And in [326], using Retrieval-Augmented Generation for large language models using exact nearest neighbor search, which is memory-bound, the acceleration due to AMX was from 2 to 10%. Using Nvidia H100 GPUs, the acceleration was from five (with one H100) to forty times (with 8 H100), although [326] noted the high cost of hardware with H100 (8 GPUs were required for the version with a memory capacity of 512 GB). But the benefits of Xeon SPR using AMX are clear, and Xeon EMR provides a further significant improvement.

Intel believes that AI inference workloads will be used by little bit companies for lightweight AI models (rather than massive models with over a trillion parameters) that may only require a few cores on a modern Xeon processor [327]. This is supported by Intel's development of OpenVINO software [328], which is focused on high performance primarily for inference.

Another illustration of where non-GPU servers with Xeon processors can be effectively used for AI is the AI-oriented MLPerf-Inference benchmarks [215] (here we mean modern results of these benchmarks). These benchmarks have become a standard and are very actively used in recent scientific research in the field of AI. The relevant results are presented in [329].

In the server scenarios of these benchmarks (for online applications with random requests, where the overall latency target is taken into account when generating result [329]), there are results provided by Intel, where a 2P server with 64-core Xeon 8592+ without GPU outperformed the results with modern Nvidia GPUs. For example, in the dlrm-v2-99.9 recommendation model benchmark (the last number is the accuracy level in percent), the Intel server gave 949.2 requests/sec, while the Nvidia data with the H200-SXM-141GB-CTS GPU indicated 155.0 requests/sec. The number of requests/sec here indicates the number of requests processed while meeting the latency bound requirement.

In the large language model benchmark (gpt-j-99.9 in the server scenario) with a Xeon 8592+ Intel, the highest result was obtained on

14.01.2025, 949.20 tokens/s, and in one of the results from Supermicro with 8 Nvidia H100-SXM-80GB GPUs — 900.0 tokens/s. In late summer 2024, Nvidia announced that it had obtained significantly higher results for such AI inference benchmarks [**330**], but the corresponding results were not yet available on the MLperf testing site at the time of writing.

Throughput is a metric for offline scenarios in MLPerf-Inference benchmarks [**215**] (for batch processing applications where all data is available immediately and latency is unlimited). Performance data for Xeon 8592+ servers for offline and server scenarios of Llama-2b-99 and Llama-2b-99.9 under MLPerf-inference benchmarks conditions is available in [**329**]. For example, in the benchmarks set version 4.1 there is a result obtained by Intel for the server variant of Llama-2b-99 on a server with Xeon 8592+, which is 8% higher than the result obtained by Dell for a server with 8 AMD MI300X GPUs (data as of 14.01.2025).

Dell's performance data for its servers with Xeon EMR in MLPerf-Inference 4.0 is available in [**331**], and for Xeon SPR in large language models using FP32, FP16, and BF16 formats is available in [**332**].

It is clear that for the above inference tasks, Xeon EMR processors are relevant. Another illustration of this can be considered the work [**333**], which proposed an improved approach to accelerating the inference of large language models using parallelism, as a result of which, with acceptable values of achieved throughput and latency, using Xeon EMR 6538N, energy consumption was reduced by 49% compared to using a server with Nvidia A100 GPU.

While the above are isolated examples of AI tasks where Xeon performance successes over GPU systems are demonstrated, this is of course only an illustration of the fact that such tasks exist. Of course, on a huge number of other AI benchmarks that typically require more compute resources, x86 processors, including Xeon, lag far behind modern GPUs in performance.

After AMD provided a video at Computex 2024 showing that its latest 128-core Turin processors (Zen 5 architecture, see Section 6 below) outperformed the 64-core Xeon 8592+ processors in large language model inference throughput without specifying the software stack or service level agreements (SLAs) used, Intel showed the opposite results [**334**].

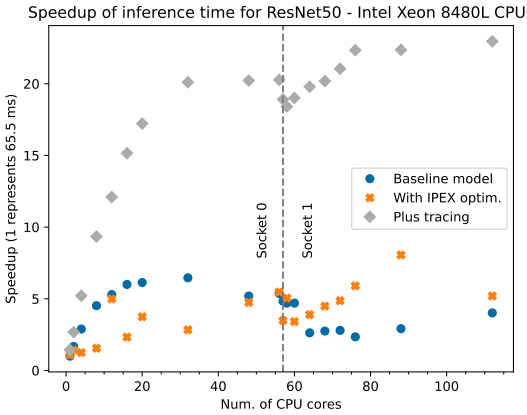


FIGURE 36. Inference times for the ResNet-50 model. Base times are shown in blue, and those obtained using IPEX are shown in yellow (figure from [335])

Both information appeared before AMD started shipping these Zen 5 Turin processors. On a 2P server with a Xeon 8592+, using Llama2-7B with INT4 format, 50ms SLA at the P99 percentile (i.e. when only 1% of requests are processed in a time greater than the latency) using Intel's open source PyTorch extension (IPEX), a throughput of 686 tokens/s was achieved versus 671 for a 2P server with 128-core Turin processors.

Intel also showed in [334] much higher INT8 inference throughput for several deep learning models on 2P servers with Xeon 8592+ and Xeon 8480+ compared to 2P servers with more cores on EPYC 9654 and 9754.

In [335], interesting data were obtained on the reduction (with increasing number of used cores) of inference time for ResNet-50, Yolo [336] and OpenVINO models when using IPEX on a 2P server with 56-core Xeon 8480+ (see Figure 36).

Table 38 shows the PTS benchmarks data for servers using Intel's oriented to performance of inference OpenVINO software.

The latency benchmarks results are not shown here, since they often show higher results on simpler (non-server) x86 processors. The author has no data on the use of AMX on Xeon processors in these benchmarks.

TABLE 38. Performance (results/sec.) in OpenVINO 2024.0 inference PTS benchmarks

	N	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Xeon 8592+	2P	5338		1130	**546**	**32006**	
	1P	2808	2597	592	**284**	**16716**	406
Xeon 8490H	2P	4093	4396	796	425	28502	653
	1P	1902	1541	477	168	8944	299
Xeon Max 9480	2P	2940	3567	812	380		535
	1P	1804	1541	476	218		301
EPYC 9684X	2P	**8279**	5196	1112	206	10610	**1107**
	1P	**4139**	2553	583	99	4972	**475**
EPYC 9654	2P	7812	5032	1015	197	10237	767
	1P	3185	**2609**	546	96	5007	352
EPYC 9754	2P	6474	**5824**	**1146**	241	12294	759
	1P		2504	**621**	122		277

Data from [https://openbenchmarking.org/test/pts/openvino] as of 17.01.2025.
Average values of different executions of tests are given.
Maximum values for the given processors are shown in bold.
N is the number of processors in the server.
Test 1 — Vehicle Detection. Format: FP16
Test 2 — Handwritten English Recognition. Format: FP16
Test 3 — Machine Translation EN to DE. Format: FP16
Test 4 — Face Detection. Format: FP16-INT8
Test 5 — Weld Porosity Detection. Format: FP16
Test 6 — Person Detection. Format: FP16

Although in some of these benchmarks the high-end Xeon EMR and Xeon SPR models outperformed the EPYC 9004, the latter were more often ahead (see also Table 22). Other data from the OpenVINO 2023.2.dev PTS benchmark for such Xeon models are given in [**202**]. There, too, in some cases the servers with Xeon 8592+ were faster, in others — with the high-end EPYC 9004 models.

It should also be noted that one should be quite careful in assessing the meaning of the numerous x86 performance data that appear in a wide variety of AI inference benchmarks. As an example, we will point out the inference performance data (frames per second) for the usual

TensorFlow framework benchmarks (for the high-end models of the processors considered in the review)[43].

Similar performance data is available, for example, in [**200**] and [**289**]. For such benchmarks using the ResNet-50 model when working with batch sizes (BS) of 64 and 256 (there is also data for Xeon 6 and EPYC Zen 5), situations arise, when replacing a processor model with fewer cores to an high-end model with more cores, or when moving from a 1P to a 2P server configuration, the performance drops or scales poorly.

Report [**200**] provides data from such benchmarks at BS=512, but similar data[44] also contain, perhaps weaker, effects of poor scalability with the number of cores. Roughly speaking, in such benchmarks, higher performance can be achieved in servers with cheaper and having lower performance processors with fewer cores. Accordingly, comparing the performance of such processors in these benchmarks has very limited meaning.

Comparison of the considered processors by the latency values in such benchmarks is less interesting. It is clear that in such benchmarks it would be necessary to conduct a study of the dependence of performance on the number of cores used.

A large number of benchmark results for the performance and energy efficiency of a 2P server with Xeon 8592+ for inference and training, using a wide range of AI models, different frameworks and number formats, for different batch sizes are presented by Intel in [**337**], and for a 2P server with 56-core Xeon 8480+ in [**338**]. A single processor was used for training there. Similar data can also be found there for 32-core Xeon SPR 6448Y, and for 3rd generation Xeon Scalable processors, as well as for the Intel Flex Series 140 and 170 GPUs. These data allow, in particular, to obtain estimates of the performance progress for AI tasks when moving to newer generations of Xeon Scalable processors.

Intel presented such data on the performance acceleration in a 2P server with Xeon 8592+ compared to Xeon 8480+ for inference and training tasks when using the TensorFlow framework for different AI models with BF16 and INT8 formats and with different BS [**339**]. BS=1 in Figure 37 for inference corresponds to the use of real time, and BS=x means the use

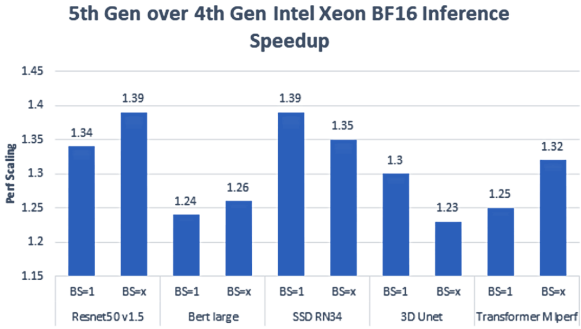


FIGURE 37. Speedup on Xeon 8592+ vs. Xeon 8480+ in AI inference (figure from [**339**])

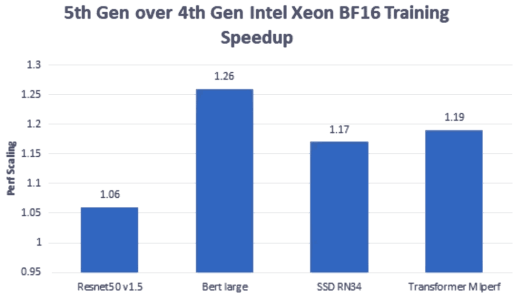


FIGURE 38. Speedup of Xeon 8592+ vs. Xeon 8480+ for AI training (figure from [339])

of the performance-optimal batch size. The inference acceleration when working with Xeon 8592+ compared to Xeon 8480+ is from 23 to 39%.

Figure 38 shows the performance improvement of the Xeon 8592+ over the Xeon 8480+ for the models training with mixed-precision BF16/FP32 case, with speedups ranging from 6% to 26%.

An example of the efficiency of Xeon EMR for AI in terms of power consumption was already given above. In [**340**], a study of the energy efficiency of various AI models was conducted for a 2P server with 64-core

[43] https://openbenchmarking.org/test/pts/tensorflow, accessed 4.12.2024.

[44] https://openbenchmarking.org/test/pts/tensorflow, accessed 4.12.2024.

Xeon 8592+, including large language models, the ResNet-50 image recognition model, and the Bert-Large natural language processing model.

This was achieved by using the capabilities of the Xeon EMR's enhanced OPM optimized power mode feature, which uses subtle tools including changing the frequency of the internal interconnect. OPM's operation resulted in an increase in energy efficiency of up to 25%, but the improvement is only possible in the case of low processor load (up to 50%).

In conclusion of the consideration of the performance of Xeon SPR and Xeon EMR in AI tasks, we will point out the data from the work [**311**], which is initially focused on the creation of portable software kernels for different hardware platforms using Tensor Processing Primitives (TPP) common for HPC and AI tasks. It presents, in particular, data on the performance of 2P servers with 56-core Xeon 8480+ and 64-core ARM AWS Graviton 3 (with Neoverse V1 cores) for matrix multiplication tasks, which are of interest primarily for deep learning (GEMM in BF16 and FP32 formats).

The corresponding data are presented in Figure 39; for different software variants of GEMM (PARLOOPER refers to the implementation in [**311**]), they demonstrate, in particular, a large performance advantage of Xeon SPR processors over ARM with a comparable number of cores.

The Xeon SPR and especially the Xeon EMR processors are certainly interesting for AI applications. They have certain performance advantages over the EPYC 9004 due to hardware features (primarily due to AMX support) and Intel's powerful AI software tools (a number of which have already been mentioned).

There is a lot of data that demonstrates the performance advantage of servers with Xeon 8490H or Xeon 8592+ over servers with high-end EPYC 9004 models for AI tasks, for example, in the PTS benchmark OneDNN 3.4 with the Harness: Deconvolution Batch shapes_3d configuration[45], with OneDNN 3.3 in [**202**]. In such OneDNN benchmark, the selected configurations should also be controlled, since they may provide inappropriate scaling with the number of cores for such processors.

[45] https://openbenchmarking.org/test/pts/onednn, accessed 6.02.2025.

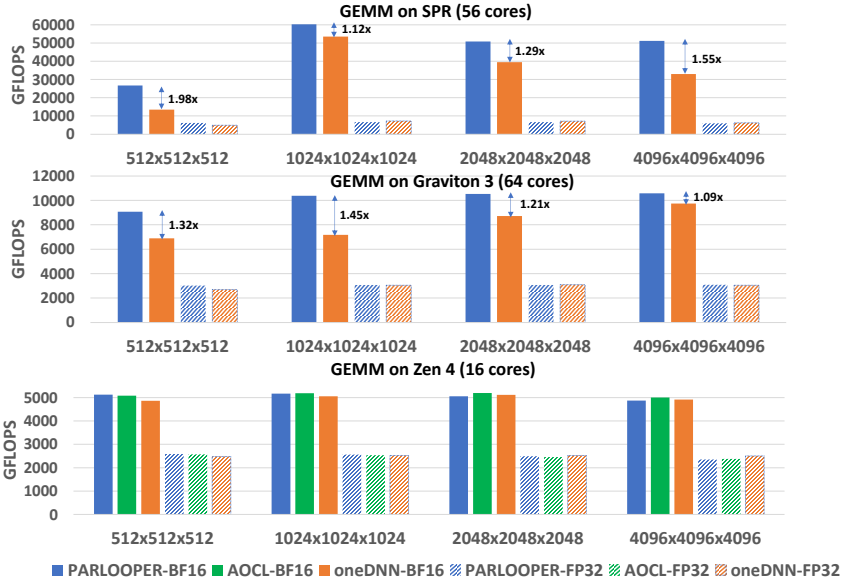


FIGURE 39. Performance of different GEMM implementations (figure from [**311**]); data "GEMM on Zen 4" do not relate to the server processors discussed here

Here and earlier examples were given where the high-end EPYC 9004 models outperformed the corresponding Xeon SPR or EMR models in AI tasks. We can also point to AMD data [**341**] on the higher performance of a 2P server with an EPYC 9654 compared to a server with a Xeon EMR 8592+ when using XGBoost (eXtreme Gradient Boosting) — a scalable high-performance method for machine learning when applied to datasets for predicting flight delays or detecting Higgs bosons.

However, many of these results do not include data on the use of Intel software optimized specifically for the new Xeon processors. In any case, it is clear that for AI tasks, the competitiveness of Xeon SPR and especially Xeon EMR compared to EPYC 9004 is greater than in other widely used application areas (discussed above in Section 4.1).

4.3. Comparison of Performance of Individual Types on Xeon SPR

In this section is carried out performance comparisons between Xeon SPR and EPYC 9004 on midrange models, as well as performance

comparisons between Xeon SPR and GPGPU.

Performance of conditional middle-class models Xeon SPR and EPYC 9004. At the University of Utah HPC Center, calculations were performed in various benchmarks on servers with different EPYC 9004 and Xeon SPR models [**100**], and it was shown that among mid-range models, the performance of Xeon SPR (32-core Xeon 6430) is usually slightly inferior to the similar EPYC 9004 (32-core EPYC 9334), but has a significantly lower price. In addition to comparing servers with mid-range models, benchmarks were also performed on 1P configurations using high-end EPYC 9004 models, including those with the same number (64) of cores as in 2P models.

The focusing [**100**] on mid-range models was due to the assumption that they may be more actual for numerous HPC calculations, including in small clusters. And, in addition, the available mass data of PTS test results on the openbenhcmarking.org website mainly relate to the high-end models of these processors. It can be assumed that the mid-range processor models in [**100**] are those priced around $3000 and below.

In [**100**], interesting in this regard benchmarks were conducted, including HPL, HPCC, NPB, and computational chemistry applications (LAMMPS, GROMACS, and NWChem). A certain disadvantage of these data is the use of the Spack package manager [**342**], which is primarily oriented toward HPC and supercomputers, with parameters for constructing executable modules that are not narrowly oriented toward the corresponding processor models.

Some data from [**100**] are given in other sections of the review; here we will limit ourselves mainly to the HPCC benchmarks results presented in Table 39.

From [**100**] it is clear that these Xeon and EPYC models with the same number of cores show competitive performance data. In some benchmarks, the Xeon 6430 surpassed the EPYC 9334 in performance, including in HPL and DGEMM. However, when using the MKL library (for the EPYC 9334—with the mode enabled that provides maximum performance), in HPL the obtained performance with the EPYC (3.53 GFLOPS) was higher than with the Xeon 6430 (3.44 GFLOPS).

TABLE 39. HPCC benchmark performance data

Model		Xeon 6430	EPYC 9334
Number of cores		2×32	2×32
Base frequency, GHz		2.1	2.7
Price		$2138	$2990
TDP, W		270	210
HPL, TFLOPS		3.23	3.11
DGEMM Single GFLOPS		71.14	59.87
PTRANS GB/c		18.20	29.85
Random Access, GUPs	Single	0.107	0.148
	MPI	0.337	0.445
FFTE MPI, GFLOPS		57.73	89.58
Stream triad, Single GB/c		13.28	43.20

Data from [**100**]. Single refers to tests on a single core.

You can also pay attention to the SPEC CPU2017 fp_speed and fp_rate [**51**] benchmarks data. As of 02/13/2025, the maximum results achieved for 2P servers with EPYC 9334 were 339/358 and 843/845 for fp_speed and fp_rate, respectively (base/peak values are given). Similar results for Xeon 6430 are 295/295 and 683/703, respectively. For 1P servers, EPYC 9334 also turned out to be faster in these benchmarks.

The Xeon 6430 has a significantly lower cost, although it has a higher TDP than the EPYC 9334. From the above data, we can assume that the Xeon 6430 model is more competitive (in relation to EPYC 9004 processors) compared to the high-end Xeon SPR models.

In Table 40, we present the SPECcpu 2017 floating point benchmark results for mid-range Xeon ICL/SPR/EMR and EPYC Genoa processors. The 64-core EPYC 9554P is included here as an illustration of a possible alternative to two 32-core EPYC Genoa processors. All processors in the table (except for this 64-core processor) can be classified as mid-range processors with about 30 cores and a price of less than $3,000. Accordingly, the lowest-end 32-core EPYC Genoa model is presented here (the 32-core EPYC 9354 model costs more than $3000).

Table 40. SPECcpu 2017 benchmarks data for EPYC 9004, Xeon ICL, Xeon SPR, and Xeon EMR

CPU	EPYC 9334		EPYC 9554P	Xeon 6330		Xeon 6430		Xeon 6530	
Number of CPUs	1	2	1	1	2	1	2	1	2
Number of cores	32	64	64	28	56	32	64	32	64
Price	$2990	$5980	$7104	$2128	$4256	$2138	$4276	$2128	$4256
SPECcpu 2017 fp_speed peak/base	253/239[1]	358/339[2]	308/301[3]	116/114[4]	219/219[5]	186/186[6]	295/295[7]	–/220[14]	326/326[15]
SPECcpu 2017 fp_rate peak/base	422/420[8]	845/843[9]	665/618[8]	188/179[10]	423/406[11]	330/318[12]	703/683[13]	377/369[6]	782/765[14]
Frequency, GHz	2.7–3.9		3.1–3.75	2–3.1		2.1–3.4		2.1–4	
TDP	210 W	420 W	360 W	205 W	410 W	270 W	540 W	270 W	540 W

Data as of 3.01.2025. Prices from [49], for Xeon 6330 — minimum of [https://www.intel.com/content/www/us/en/ark.html#@PanelLabel595 Accessed 23.02.2025]. Senders of results and systems used in calculation:

[1] ASUS RS520A-E12-RS12U;
[2] ASUS RS720A-E12-RS12;
[3] Lenovo ThinkSystem SR635 V3;
[4] HPE ProLiant DL110 Gen10;
[5] xFusion FusionServer 1288H V6;
[6] HPE ProLiant DL320 Gen11;
[7] ASUS ESC4000-E11;
[8] xFusion FusionServer 1158H V7;
[9] xFusion FusionServer 2258 V7;
[10] HPE ProLiant DL110 Gen10 Plus;
[11] Tyrone Camarero TDI100C3R-212;
[12] Dell Precision 7960 Rack;
[13] ZTE R5300G5;
[14] Lenovo ThinkSystem SD530 V3;
[15] xFusion FusionServer 1288H V7;
[15] ASUS RS720-E11-RS12U.

According to this table, among mid-range processors, EPYC outperforms Xeon in terms of performance with the same number of cores. Although Xeon may be better in terms of cost/performance, this requires additional analysis. In addition, TDP values should be taken into account.

Comparing Xeon SPR performance with GPGPU. Finally, in this section, it is useful to also provide data on the relative performance of Xeon SPR on HPC applications compared to modern GPGPUs to assess the possible efficiency of their use. For this purpose, we use data for a server with two Xeon 8480+, to which Nvidia A100, H100 and H200 GPUs were added [**343**] (data as of 10/9/2024). We selected data for two well-known HPC applications in the field of CFD (FUN3D) and molecular dynamics (GROMACS). Similar data are presented in [**343**] for other molecular dynamics programs (AMBER, NAMD and LAMMPS), with calculations of different molecular systems. GROMACS was chosen here due to the frequent perception of this application as having the high performance and achieving better parallelization.

When adding one A100 GPU to the server, FUN3D performance in the dpw_wbt0_crs-3.6Mn_5 benchmark increased 4 times, and in GROMACS in the STMV benchmark for a molecular system containing about a million atoms — 2 times. When adding the most modern H200 GPU instead of the A100, the performance of these applications increases another two times. It is also interesting that when using instead of the server the Nvidia GH200 version, the performance practically does not change [**344**], that is, using the integrated Grace processor in this regard is no different from using a pair of Xeon 8480+.

According to [**344**], in the NAMD application in the Her1-Her1 test, the performance of a 2P server with 60-core Xeon 8490H was more than four times slower than the performance of a single Nvidia Geforce RTX 6000 Ada GPU.

Comparison of the energy efficiency [**345**] of a server with a Xeon 8480+ relative to the use of an Nvidia H100 GPU or an integrated GH200 on the NAMD molecular dynamics application or on the MILC software for the theory of strong interactions in subatomic physics certainly shows the advantage of using a GPU. However, all this can only serve as a numerical illustration for assessing the feasibility of their use if such a possibility exists.

Here are three more examples comparing Xeon SPR and GPU. In [346], the performance of a 2P server with 56-core Xeon 8480+ and Intel Max 1550 GPUs in the OpenFOAM CFD application is compared. The performance of another 2P server with Xeon 8480+ for CFD tasks using OpenFOAM and AI tools when working with different numerical formats and using mixed precision compared to a single Intel Max 1550 GPU is compared in [347]. And in [335], the inference time of a 2P server with Xeon 8480+ or with Intel Max 1100 GPU in AI models, including ResNet-50, is analyzed with the dependence on the number of cores used in Xeon.

In conclusion of this subsection, it should be noted that achieving performance gains by using a GPU may require a lot of work and is therefore not always realized. For example, in [283] in the magnetohydrodynamics application MHD, a 2P server with a Xeon SPR with HBM memory (Xeon Max 9480) yielded performance close to that of an Nvidia A100 GPU. The performance of the Xeon Max is discussed below.

4.4. Xeon Max Processors Performance

A good overall demonstration of Xeon Max performance relative to Xeon ICL and Xeon SPR is Figure 32 above, which compares data for 2P servers with 56-core Xeon Max 9480, Xeon 8480+, and 40-core Xeon 8380.

Earlier in the review, in the process of examining data on the performance of processors of other architectures, data on the performance of Xeon Max was repeatedly cited, including in a relative form (of a comparative nature).

The advent of x86 processors with HBM memory support (Xeon Max are the second processors in the world after the A64FX to support HBM and the first to support both HBM and DDR) during the modern expansion of performance limitation in computing systems due to memory bandwidth has generated various hopes and caused a natural surge in the number of publications on Xeon Max performance in various fields of science (see, for example, [3, 163, 278, 283, 284, 348–351]), including the works of Intel employees themselves [352]. Of particular note is the publication that examines the use of Xeon Max in nodes of the Aurora supercomputer and provides information on the performance of HPC applications [353].

Some of the expectations that arose were not met, although they could potentially have been clearly overstated. For example, in [348] it was pointed out that HBM memory was insufficient to completely overcome the performance gap at thermonuclear fusion simulations between CPUs and GPUs.

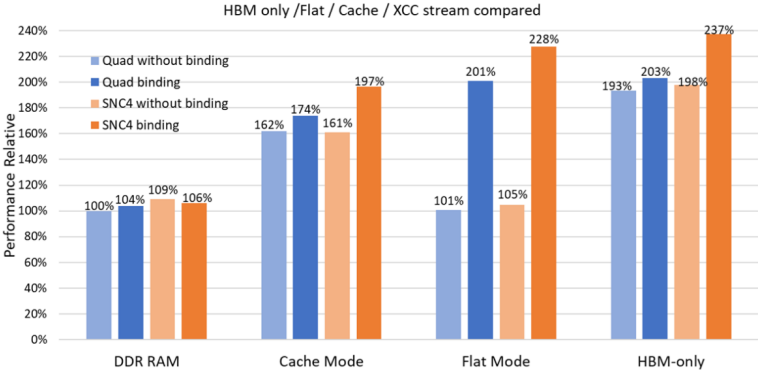


FIGURE 40. Relative memory bandwidth of Xeon Max 9480 in different modes (figure from[280])

Possible reasons for the limitation in the expected performance growth are discussed below. The main issues here are the achievable memory indicators (primarily bandwidth), since HBM is quite expensive (see prices for different Xeon Max models above in Table 26).

It is clear that after the Xeon Max entered the market, studies of the achieved latencies and memory bandwidth of HBM immediately began to appear; see, for example, [**284**]. An analysis of the dependence of the latency and bandwidth of HBM in a `1P` configuration with Xeon Max 9470C on various memory configurations is carried out in [**353**]. We will further focus on the most important indicator, bandwidth, which is usually measured in STREAM benchmarks.

Xeon Max memory bandwidth in STREAM benchmark. The impact of different HBM memory modes on the bandwidth in a `2P` server with 56-core Xeon Max 9480s relative to the bandwidth in a `2P` server with 60-core Xeon 8490Hs connected only to DDR5 memory was shown in [**280**] and is presented in Figure 40. Most likely, the STREAM Triad variant was used there, although this was not explicitly stated in [**280**]. This figure also illustrates the impact of NUMA domain affinity, which for Xeon Max helps to increase the achieved bandfwidth (quad here means operation with quadrant, when all the HBM memory of the Xeon Max processor forms one NUMA node).

The cache mode showed the lowest bandwidth as expected, and the HBM-only mode with `SNC4` showed the highest bandwidth. In the combination of Flat mode with quadrant clustering, the dependence on the number of cores involved, from 2 to 56, was investigated in [**280**]; the

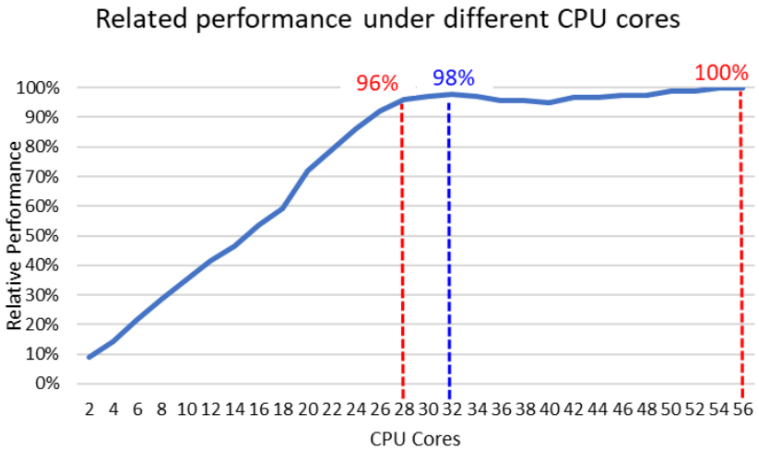


Figure 41. Dependence of the achieved memory bandwidth on the number of cores used (figure from [**280**])

corresponding data are presented in Figure 41. They show that good scaling of bandwidth is observed only up to 28 cores, reaching 96% of the maximum obtained value.

Logarithmic scaling of the number of cores in the similar results in Figure 33 (in Section 4.1) probably does not allow one to see a similar [**280**] bandwidth dependence.

It should be noted that, according to data from [**283**], the maximum bandwidth when working with DDR5 was almost achieved with 7 cores.

The bandwidth in the STREAM Triad benchmark for a 2P server with 48-core Xeon Max 9468 in the actual, allowing higher performance, Flat mode with SNC4 was measured in [**14**]. The value obtained there, about 1361 GB/s, turned out to be 1.6 times higher than for a 1P server with ARM A64FX.

An important detailed study of the specifics of working with HBM memory in Xeon Max, including bandwidth in STREAM benchmarks, is conducted in the works of the author of the STREAM benchmarks themselves, McCalpin [**281**, **354**].

An illustration of this is Figure 42, which shows that in the STREAM benchmark, even ideal for bandwidth parallelization on Xeon Max cores would only achieve 68% of the peak 1638 GB/s for HBM.

This data refers to a 1P server with a 56-core Xeon Max 9480. Accordingly, the memory bandwidth problem was researched at the core

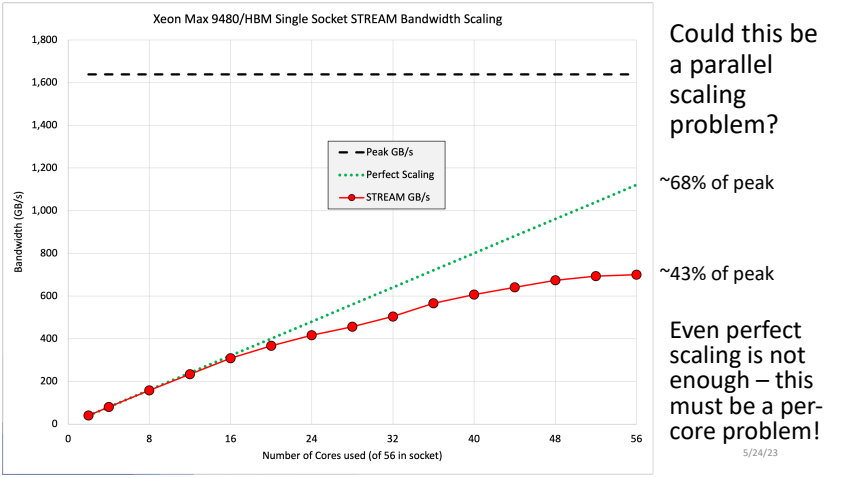


FIGURE 42. HBM bandwidth in Xeon Max 9480 depending on the number of cores involved (figure from [354])

level. And the parallelization itself, as the data in this figure show, is far from ideal.

A study of possible parallelization in processing L1 and L2 cache misses, taking into account the latency according to Little's law from queueing theory, yielded a maximum bandwith of about 24 GB/s for a single core, and measured values showed 20 GB/s in STREAM and 22 GB/s in the "read only" from memory benchmark. But the actual measured bandwidth of the processor (on all cores) was much less than predicted by this theory.

A detailed, very subtle analysis of Xeon Max indicators during benchmarks runtime showed that the 2D mesh interconnect in these processors provides only half the bandwidth needed to achieve the full possible read bandwidth from HBM [281, 354].

Despite achieving less than half the possible HBM bandwidth due to the existing limitations of parallelism in processing cache misses in Xeon Max cores and the bandwidth of their interconnect, Xeon Max processors provide higher HBM bandwidth compared to DDR by 2.5 to 3.5 times (in the read-only and STREAM triad benchmarks, respectively) [281]. So what is important is the real-world performance gains achieved at higher cost due to the use of HBM.

Following the publications of [281, 354], some other works also noted certain disadvantages of using HBM in Xeon Max. Thus, in [283] the low effective bandwidth per core was pointed out.

For 1P and 2P servers with Xeon Max 9480 in the BabelStream Triad benchmark, the dependence of the achieved bandwidth on the size of the arrays used in testing (from 19.2 MB to 9.8 GB) in the HBM-only mode was researched in [**163**]. The maximum bandwidth was achieved with sizes no more than a few hundred megabytes, and then it decreased. It is possible to talk about the memory bandwidth only with sufficiently large array sizes, when the measured value has stabilized; with smaller sizes, the obtained value refers to the cache operation [**163**].

The corresponding stabilized value for the 2P server was 1446 GB/s when using the compilation flags optimal for other studied tests, and 1643 GB/s when using a special STREAM-optimized set of flags (streaming store) from Intel OneAPI Base & HPC toolkits 2023.1. These values are 55% and 63% [**163**] of the peak bandwidth, respectively, for which the value from the processor's capabilities was used, taking into account cache memory latencies — 1.3 TB/s per socket [**354**]. The obtained percentage values correspond to the above conclusions [**281**, **354**].

AMD in [**160**] demonstrated the composite average values of bandwidth from all four STREAM components for 2P servers with 32-core Xeon Max 9462 and EPYC 9384X with 3D V-cache for different numbers of used threads from 1 to 64 as relative indicators. In all used numbers of threads (powers of two), the bandwidth in EPYC was higher than that of Xeon Max in Cache and HBM-only modes, except for the latter at 64 threads. However, [**160**] does not indicate the sizes of the arrays used, which may mean that the bandwidth advantage may be achieved not by the memory itself, but by 3D V-cache.

Xeon Max performance in SPEC benchmarks. Work [**355**] studied the impact of HBM on Xeon Max performance in SPECcpu 2017 and SPEChpc 2021. Only the officially validated SPEC benchmark results are considered here.

The performance data for Xeon Max processors in the SPECcpu 2017 fp_speed and fp_rate benchmarks is shown above in Table 27. Here we compare the performance of Xeon Max with Xeon SPR and Xeon EMR with the same number of cores. With a higher number of cores, Xeon SPR or EMR clearly outperform Xeon Max in these benchmarks.

For 2P servers with the same number of cores (56 per processor), switching from the Xeon 8480+ to working with HBM in the Xeon Max 9480 increases the performance in the peak SPECfp_rate indicator by 4%. Perhaps due to the slightly lower core frequencies of the Xeon Max 9480, it is inferior to the Xeon 8480+ in SPECfp_speed. And the 56-core Xeon

8570 without HBM memory has higher performance in both SPECfp 2017 benchmarks at a lower cost, so the progress in Intel technology and DDR memory technology turned out to be much more important for Xeon EMR in this regard than the transition to expensive HBM memory. In the 5th generation of Xeon Scalable processors, as in the Xeon 6, Intel is not offered HBM memory.

The Xeon Max performs have a lower performance than the high-end EPYC 9004 models in these benchmarks. The SPECcpu 2017 integer arithmetic results are not of interest for the purposes of this review. The data does not change the picture of the relative performance of the Xeon Max and EPYC 9004.

More interesting is the performance of Xeon Max in SPEChpc 2021. For servers and clusters with Xeon Max 9480 for all benchmark variants from tiny to large, the corresponding data is presented on the website[46]. In this data as of 1.04.2025, 2P servers with Xeon Max 9480 are inferior in performance to 2P servers with EPYC 9654.

The Xeon Max performance data that is actual to this review and discussed below pertains to only two models, the 48-core Xeon Max 9468 and the 56-core Xeon Max 9480 (data for these models is provided in Table 26 above). Publications that considere performance in a broad class of benchmarks or applications, only by one of these models per article usually is discussed. Therefore, for convenience of presentation, the performance analysis below is also divided into two parts for different models. Each of the corresponding sections includes different benchmarks. And the second of these sections also provides the necessary generalization.

4.4.1. *Performance of 48-core Xeon Max 9468*

Since Xeon Max has HBM memory, which was previously used in A64FX, publications have appeared comparing computing systems based on them. Thus, in [**356**] for deep learning (for multilayer perceptron), the performance and power consumption in Infiniband clusters with 2P nodes based on Xeon Max 9468 and 1P nodes with A64FX were investigated, and their scaling depending on the number of cores and cluster nodes was studied.

In general, there is some interesting data on the performance of the Xeon Max 9468, which includes comparisons with various ARM processors (see Table 41).

[46] https://www.spec.org/cgi-bin/osgresults?conf=hpc2021, accessed 1.04.2025.

Table 41. Xeon Max 9468 performance data compared to ARM processors

Benchmark (units of measured performance)[1]	Input Data	Intel Xeon Max 9468, 96 cores[2]		Fujitsu A64FX, 48 cores	Amazon Graviton 3, 64 cores	Nvidia Grace, 144 cores[3]
		DDR	HBM			
DGEMM (GFLOPS)		4787	5392	1978	1/158	4461
HPL (GFLOPS)		2211	2862	1177	965	3124
FFT (GFLOPS)		129	143	24.4	71.0	134.2
HPCG (GFLOPS)		84	198	64.4		106.5
OpenFOAM (execution time, minutes)	MotoBikeQ, 11M cells	18.39	14.87			13.87
	MotoBikeX6, 35M cells	53.45	39.65			
GROMACS (ns/day)	MEM 82K atoms	203.6	206.1	22.8	71.4	171
	RIB 2M atoms	13.9	13.5			12.7
	PEP 12M atoms	1.2	1.2			0.977

The table uses data from [14, 357].

[1] The first 3 benchmarks listed in the table are taken from HPCC.

[2] The explanation of the preferred HBM variant and DDR is given in the text below.

[3] The maximum values presented in [357] obtained using different software tools are given.

As for the preferred HBM variant used for the Xeon Max 9468 benchmarks, it was set by specifying the numactl flag, -preferred-many, for all HBM NUMA memory domains. As noted in [**14**], this gave the same performance in their benchmarks as using NUMA domain affinity.

While some of the data in this table is for 1P servers (with A64FX, Graviton 3), the Xeon Max 9468 showed performance advantages over all modern ARM processors (although in some cases this required heavy use of HBM). The same is true for per-core performance (calculated based on the data in this table).

The exception was the OpenFOAM benchmark, where the performance of the Nvidia Grace processor (using 32 LPDDR5X channels), which has one and a half times more cores, was 7% higher than that of the Xeon Max 9468 with HBM. In terms of performance per core, the Xeon Max is ahead here. It should also be taken into account that with good performance scaling in the benchmark, the Xeon Max 9480 should show higher results, and that there are having more high-performance models in the Xeon SPR. And in the Xeon EMR, an additional increase in performance is achieved.

For AI tasks (in the AI Benchmark Alpha, not shown in the table due to lack of data for the Xeon Max), even the older Xeon ICL 6330 usually outperformed ARM processors in performance [**357**].

Table 41 shows how the higher HBM memory bandwidth improves performance on compute-intensive benchmarks (e.g. DGEMM), and how this effect is amplified on memory-bound benchmarks (e.g. HPCG).

If we compare the performance of the 48-core Xeon Max 9468 in DGEMM and HPL with the performance of the EPYC 9004 (see Table 15), we can see that this Xeon Max model is ahead of the 32-core EPYC 9004s, and lags behind the 64-core EPYC 9554, and even more so behind EPYC processors with a higher number of cores. In the requiring high bandwidth HPCG benchmark with the efficiently used HBM memory variant (197.5 GFLOPS), it significantly outperforms the data presented by AMD in [**121**] (the maximum value is 137.9 GFLOPS for the EPYC 9684X).

However, for comparison purposes, it is necessary to compare the calculation data with the same input data (in the case of HPCG, for example, submesh sizes), which is unknown for the above numbers for HPCG. Accordingly, below we only consider comparable performance data in PTS benchmarks from openbenchmarking.org. It is useful to recall here that (as noted above in Section 2.4.4) the 64-core Xeon SPR 8462Y outperformed the EPYC 9004 in the HPCG 3.1 PTS benchmarks.

A number of data demonstrating the performance of the Xeon Max 9468 in various PTS benchmarks compared to the Xeon Max 9480 are provided in the following Section 4.4.2.

If you look at the PTS bencmarks data — NPB, continuum mechanics tasks where applications are often memory-bound (CloverLeaf, OpenFOAM, WRF), molecular dynamics tasks (NAMD, LAMMPS, GROMACS) and molecular docking (miniBUDE) on the openbenchmarking.org website, then according to data as of December 2024, the Xeon Max 9468 is almost always inferior in performance to the high-end EPYC 9004 models.

But the situation is different in the PTS benchmarks for AI tasks. In the OpenVINO benchmark variants that can be classified as image processing (data as of 17.01.2025), the Xeon Max 9468 often outperformed the 96-core EPYC 9654 and 9684X, although it sometimes lagged behind the 128-core EPYC 9754.

In the TensorFlow PTS benchmark with the ResNet-50 model as of 18.01.2025, the Xeon Max 9468 outperformed the EPYC 9004 with a batch size of 256, but lagged behind with larger batches (512). In some oneDNN PTS benchmarks (as of 19.01.2025), the Xeon Max 9458 also outperformed the 96-core EPYC 9004s, but lagged behind the 128-core EPYC 9754.

However, these PTS benchmarks mentioned in the three paragraphs above often have poor performance scaling with the number of cores used. This means that performance decreases when comparing processors with fewer cores compared to high-end models with more cores, or when moving from 1P to 2P configurations. This reduces the value of the benchmarks in question.

Since the next section will discuss the performance of the Xeon Max 9480 versus the Xeon Max 9468, we will conclude our analysis of this model here.

4.4.2. *Performance of 56-core Xeon Max 9480*

It is clear that a lot of performance data has appeared for the top model of this family, Xeon Max 9480. Thus, in [**283**] it is shown that a 2P server with Xeon Max 9480 in DGEMM and HPL benchmarks provides only 87% of the performance of a 2P server with also 56-core Xeon 8480+.

Other data comparing the performance in HPL of a 2P server with Xeon Max 9480 and similar servers with Xeon SPR and Xeon EMR are available in [**303**]. There, 6781 GFLOPS were obtained with Xeon Max 9480, and with Xeon 8480+ higher, 7293 GFLOPS (it is clear that the 60-core top model Xeon SPR 8490H achieved even higher performance, 7584 GFLOPS). The corresponding performance data for the Xeon Max 9480 compared to other Xeon SPR models is presented above in Table 31.

This performance lag from the Xeon 8480+ is obviously due to its cores having higher frequency (both base and turbo) than the Xeon Max 9480, and illustrates that the Xeon Max may only be relevant for memory-bound applications.

In the HPCG 3.1 PTS benchmark with submesh sizes 144 on a 2P server with a Xeon Max 9480 in [**358**], a performance of 74.3 GFLOPS was obtained, which, however, is lower than the data in Table 41, obtained for the Xeon Max 9468.

And in the S-FFTE benchmark, where memory bandwidth is important, the server with the Xeon Max 9480 was faster than the server with the Xeon 8480+ by a factor of 1.46. And in memory-bound magnetohydrodynamics tasks, the performance of the Xeon Max 9480 was much higher [**283**]. In general, in the CloverLeaf 2D/CloverLeaf 3D magnetohydrodynamics mini-application, the performance of the Xeon Max 9480 is much higher than that of the Xeon 8480+ [**3**]. Electromagnetism is needed for thermonuclear fusion, and the HBM memory in the Xeon Max 9480 in the CGYRO application also provides a significant performance boost [**348**].

Another demonstration of the advantages of the Xeon Max 9480 for memory-bound applications working with structured and unstructured meshs is provided by the data from [**3**]. In addition to the CloverLeaf 3D and CloverLeaf 2D mini-applications just mentioned, there are also performance data for six other applications. However, all of them assume the use of specific software tools, OPS for structured meshs and OP2 for unstructured ones.

In [**3**], not only performance data but also power consumption data were obtained for various 2P servers from the Intel Developer Cloud (see Section 5 for details), including the Xeon Max 9480 (in HBM-only + SNC4 mode), Xeon 8480+ (with the same 56 cores per processor), Xeon 8592+, and Xeon 6960P (Granite Rapids). On these selected applications, the Xeon Max 9480 often outperformed not only the Xeon EMR but also the Xeon 6 in performance.

A large amount of performance data on a 2P server with a Xeon Max 9480 for a wide range of HPC applications, including weather prediction (WRF), molecular dynamics (NAMD, AMBER), quantum chemistry (PARSEC — for finite-difference calculations using the DFT method), and others, is obtained in [**355**]. There, performance is compared when disabling HBM or DDR5 memory.

A wide range of performance and power consumption data for 2P servers with Xeon Max 9480 and Xeon Max 9468 is presented in [**359**]. These data are also interesting because they examine performance in three different HBM modes: HBM-only, Cache mode, and mode with HBM memory disabled. However, instead of the Flat mode typically used for large calculations with additional DDR memory, what may require more sophisticated work, there examine performance in the case where HBM use was disabled.

Some of the data obtained in [**360**] are presented in Table 42 and Table 43.

Table 42. Performance in HPC of 2P servers with Xeon Max

Test	Xeon Max 9480 performance			Xeon Max 9468 performance		
	Without HBM	HBM Cache mode	HBM-only	Without HBM	HBM Cache mode	HBM-only
OpenFOAM, test 1[1]	4613.9	3258.6	2908.8	4629.1	3236.3	2918.4
OpenFOAM, test 2[2]	190.7	177.0	165.1	166.5	157.8	141.9
OpenFOAM, test 3[3]	237.4	199.9	176.7	245.3	195.6	177.8
LULESH, z/s	44118.3	55339.5	60681.6	45058.0	56829.2	62348.3
NPB, SP.C	140910.0	196046.2	212262.2	132417.4	184462.3	196927.9
NPB, BT.C	279271.5	328222.2	351853.6	265661.2	307050.7	330958.0
NPB, LU.C	223823.1	273779.7	278606.5	234956.3	276681.9	284194.2
NPB, MG.C	164986.7	249731.4	253047.6	162971.3	249546.4	265747.5
Quantum ESPRESSO[4]	346.0	339.1	323.7	334.5	326.0	309.1
GROMACS[5]	11.4	12.1	13.2	11.0	12.3	12.9
miniBUDE[6]	2670.9	2799.7	3032.9	3026.7	3110.1	3234.8

Data from [359]. For NPB benchmarks, performance is given in Mop/s.

[1] Execution time (seconds) in the drivaerFastback benchmark, Large Mesh Size;
[2] Mesh time (seconds) in the drivaerFastback benchmark, Medium Mesh Size;
[3] Execution time (seconds) in the drivaerFastback benchmark, Medium Mesh Size;
[4] Execution time (seconds) in the AUSURF112 benchmark;
[5] Number of calculated ns/day in the water_GMX50_bare benchmark;
[6] GFInst/s in the BM2 benchmark;

It is obvious that the performance increases when moving from DDR memory to HBM caching and then working only with HBM. It is clear that the effect should be greater for memory-bound benchmarks. However, the data in this table allows us to estimate this quantitatively. The effect of moving from the 48-core model to the 56-core Xeon Max 9480 at normal performance scaling is also obvious, but the data in the table demonstrates this quantitatively. When analyzing, it should be borne in mind that the clock frequency of the Xeon Max 9468 is 10.5% higher than that of the Xeon Max 9480 (see Table 26), and the number of cores is 17% lower.

Of the four NPB benhcmarks conducted in [**359**] with an average problem size (C), the Xeon Max 9480 in HBM-only mode, which provides the highest performance, showed an advantage in it compared to the Xeon Max 9468 in only two: SP.C and BT.C.

According to the results in Table 42, the transition from the 48-core Xeon Max 9468 to the 56-core Xeon Max 9480 in the OpenFOAM CFD-oriented benchmarks shows a small performance gain. And in the benchmark of CFD mini-application, LULESH, this transition results in a performance loss.

In the Quantum Espresso benchmark[47] used in [**359**], which was used as one of the benchmarks on the Fugaku supercomputer, the calculation was performed using the quantum chemical DFT method using pseudopotentials in the plane wave basis for a system of 112 gold atoms. However, the data in the table show that when moving from the Xeon Max 9468 to the higher-end Xeon 9480 model, the execution time in this benchmark not decrease, but increases.

And in the GROMACS molecular dynamics software suite, which is known for its high achievable level of parallelization (a paper [**361**] recently appeared on its parallelization in a cluster containing 2P servers with 64-core EPYC 9554 in nodes, a total of 65k cores), the performance gain at transition to Xeon Max 9480 was only 2%. For the data in Table 42, the widely used GROMACS benchmark, water_GMX50_bare, was used, which includes calculations of water complexes containing from 0.65 thousand to 3 million atoms.

In the molecular docking mini-application (miniBUDE), data [**359**] also shows a performance drop at transition from the Xeon Max 9468 to the top Xeon 8480 model.

[47] https://www.hpci-office.jp/documents/appli_software/Fugaku_QE_performance.pdf, accessed 10.05.2025.

Table 43. Performance (FPS) of Xeon Max in OpenVINO 2022.3 PTS benchmarks

Benchmark	Xeon Max 9480			Xeon Max 9468		
Person Detection, FP16	33.3	35.2	37.3	33.8	34.5	36.6
Person Detection, FP32	33.7	35.7	37.9	33.1	34.2	36.0
Face Detection, FP16-INT8	293.5	329.6	359.6	285.3	328.0	354.2
Weld Porosity Detection, FP16	16321.9	16480.5	17508.3	15419.6	15767.4	16960.1
Weld Porosity Detection, FP16-INT8	31507.7	30761.6	33984.7	30066.7	31679.3	33063.8
Person Vehicle Bike Detection FP16	5792.1	6157.7	6613.6	5615.0	6224.4	6539.6

Of interest for AI tasks are the benchmark data from Intel's OpenVINO tools. Many of these benchmarks were conducted in [**359**]. We excluded the latency testing from the analysis because they typically scale poorly and better results can often be obtained on desktop processors. The data selected from [**359**] are presented in Table 43.

The performance gain from replacing the Xeon Max 9468 with the Xeon Max 9480 does not exceed a few percent. Comparative performance data for the Xeon Max 9480, Xeon SPR, Xeon EMR, and EPYC 9004 are provided above Table 22 and Table 38. According to this data, the Xeon Max 9480 lagged behind the Xeon EMR and EPYC 9004 in performance, although it often outperformed the Xeon SPR 8490H.

The performance data itself of the Xeon Max 9480 in comparison with the Xeon Max 9468 model, presented above, do not indicate any shortcomings in the Xeon Max. The problem is primarily that widely used benchmarks often do not analyze the achieved performance depending on the number of cores involved in the calculations, and do not consider its dependence on the size of the object under study. Although one cannot rule out possible problems with the level of parallelization in the Xeon Max, which were discussed in [**281**, **354**] in relation to the STREAM benchmarks. In practical terms, one can be guided by the estimate [**355**] that the increase in cost due to the use of HBM compared to Xeon SPR is compensated if the performance of a memory-bound application increases by more than 30%.

A separate comparison of the performance of Xeon Max with EPYC 9004 is not provided here, including due to Intel's discontinuation of further

support for HBM in Xeon EMR and Xeon 6. Similar performance data for these AMD processors, including a performance comparison with Xeon Max, is provided above in Section 2.4. Numerous comparative performance data for EPYC 9004 and Xeon Max for a large number of different PTS benchmarks can be found on the openbenchmarkung.org website, as well as from performance reviews that use these benchmarks (see, for example, [318]). The high-end EPYC 9004 models usually outperformed Xeon SPR and Xeon EMR in performance, and the latter, in turn, often outperform Xeon Max, with the exception of certain areas of application with memory-bound programs.

5. On the Use of x86 for Virtualization and Cloud Technologies in the Field of HPC and AI

Virtualization-based cloud technologies are also very actively used for HPC tasks, primarily, probably, due to cost-effectiveness. Cloud technologies are applicable at various levels — from individual servers and clusters to supercomputers. The use of a large number of cores in modern x86 processors exactly shifts servers with x86 to the level of possible effective use of cloud technology.

The review does hardly considered cloud technologies, but for the sake of completeness it is useful to provide general information about the existing implementations using the processors considered in the review, especially for HPC. After all, there are publications where data on performance and in the field of HPC for these processors were obtained using cloud technology.

Since there are many different types of cloud technology, we will mention only two that are widely used for HPC tasks: IaaS (Infrastructure as a Service) and BMaaS (Bare Metal as a Service). Good general recommendations for using x86 processors for different types of clouds are available for the EPYC 9004 in [221]. They can naturally be extended to other modern processors.

As examples of computing systems using such technologies, the resources of which are provided to users, we will point out the offers of Microsoft and Amazon. Microsoft Azure offers virtual machines with different processors, including those with the 4th generation Xeon Scalable processors (Xeon 8490H and 8473C) [361, 362] and EPYC Zen 4 [363] considered in the review.

Amazon also offers HPC-focused virtual machines, including with AMD EPYC Zen 4 (Genoa) [364], which the company suggests can be used for CFD and weather forecasting tasks, among other things.

Also worth mentioning is Intel Tiber AI Cloud [365], which primarily offers access to the company's cutting-edge hardware for AI tasks.

It is clear that GPUs are also actively used in servers used for cloud technologies, especially for HPC. To conclude this section, we will point out the use of Oracle cloud technologies: OCI (Oracle Cloud Infrastructure) announced new AI infrastructure capabilities for small, medium and large use cases, including the largest AI supercomputer in the cloud [366]. It actually offers BMaaS, but the computing capabilities are based on GPUs [366], and this is no longer relevant to the review.

6. About Intel Xeon 6 Granite Rapids and AMD EPYC Turin (Zen 5)

Certainly, the appearance in the fall of 2024 of Xeon 6 Granite Rapids with high-performance P-cores (high-end models, AP version — Xeon 6900P) and EPYC Zen 5 (9005 models) made a significant change in the situation with x86-64 server processors.

Already in 2025, Intel added to such Xeon 6 with P-cores an SP version — Xeon 6700P and 6500P with a smaller number of cores, a smaller number of UPI channels and a smaller number of memory channels [367], as well as with another, simpler Intel 4710 socket (the AP version uses the FCLGA7529 socket) [49]. But these processors now have the ability to build server configurations with up to 8 sockets. A series of even simpler Xeon 6300P processors also appeared [367].

Other Xeon 6 models, Sierra Forest, are not interesting for the purposes of this review, since they use simpler E-cores (without AVX-512 support), support memory with lower bandwidth and fewer channels, and have a smaller L3 cache capacity. The version with E-cores in the main Intel publication [368] at the time of writing the review is attributed to a different microarchitecture. We are considering only Xeon 6 with P-cores. Further, by Xeon 6 we mean Granite Rapids with P-cores, the AP version.

Intel began using its new 5 nm technology in Xeon 6 processors. Xeon 6 became another demonstration of progress in the direction of increasing the number of processor cores, the dominant influence on performance of the semiconductor technology used and the use of chiplets.

TABLE 44. Key indicators of Xeon 6900P and EPYC 9005

Model	Number of cores	Frequency, GHz	L3 cache capacity	TDP, W	Price	Technology
Xeon 6980P[1]	128	2–3.9	480 MB	500	$12460	5 nm
Xeon 6979P	120	2.1–3.9	432 MB	500	$11025	5 nm
Xeon 6972P	96	2.4–3.9	480 MB	500	$10220	5 nm
Xeon 6960P	72	2.7–3.9	504 MB	500	$9625	5 nm
Xeon 6952P	96	2.1–3.9	504 MB	400	$9115	5 nm
EPYC 9965[2]	192	2.25–3.7	384 MB	500	$14813	3 nm
EPYC 9845	160	2.1–3.7	320 MB	390	$13564	3 nm
EPYC 9755	128	2.7–4.1	512 MB	500	$12984	4 nm
EPYC 9745	128	2.4–3.7	256 MB	400	$12141	3 nm
EPYC 9655	96	2.6–4.5	384 MB	400	$11852	4 nm
EPYC 9565	72	3.15–4.3	384 MB	400	$10486	4 nm
EPYC 9575F	64	3.3–5	256 MB	400	$11791	4 nm
EPYC 9475F	48	3.65–4.8	256 MB	400	$7592	4 nm

[1] Data for Xeon from [https://ark.intel.com/content/www/us/en/ark/products/series/240357/intel-xeon-6.html Accessed 14.10.2024] as of 14.10.2024;
[2] Data for EPYC from [49] as of 14.10.2024.
Prices for all processors comes from [49] as of 16.04.2025.

With not very big progress in Intel's semiconductor technology, the move from Xeon SPR to Xeon EMR has not allowed the company to surpass the EPYC 9004 processors in core count and performance across a wide range of applications. Intel's move to 5nm is undoubtedly a much bigger step forward than the move from 4th-generation to 5th-generation Xeon Scalable processors.

Table 44 presents only the high-end models of Xeon 6 and EPYC 9005 Turin processors (with a higher number of cores). For example, there are a total of 26 different EPYC 9005 models with a number of cores from 8, and a number of models not in the table with a number of cores from 48 and below have a TDP of 300 W and below [126].

AMD Zen 5 now uses TSMC 4 nm technology. In terms of the number of processor cores, AMD is still ahead, but this only applies to processors based on Zen 5c cores (the latest analogue of Zen 4c), which use TSMC 3 nm technology and have a reduced L3 cache capacity per core compared to Zen 5 cores, see Table 44. To clarify the specified there technologies, we note that simpler processor blocks, such as the I/O (IOD) block, which, although with significantly different functionality than in EPYC, is now also available in Xeon 6 (see [368, 369]), use older technologies.

The large increase in the number of processor cores in Xeon 6 also required an increase in the bandwidth of the processor interconnect (there are now 6 UPI channels, with a speed of 24 GT/s — a total of 1.8 times more than in Xeon EMR [368]).

The competitiveness of the high-end Xeon 6 models relative to the EPYC 9005 should be significantly higher than it was for the 4th and 5th generation Xeon Scalable processors relative to the EPYC 9004. The Xeon 6900P was able to outperform the EPYC 9005 in some initial HPC performance data (see, for example, Table 45 below).

Analysis of microarchitectures and performance requires the availability of relevant data, which is currently being actively replenished for Xeon 6 and EPYC 9005. However, such an analysis in the review requires fairly complete data, and therefore is not performed here. At the same time, it is clear that most of what was previously discussed in this review remains valid in the latest processors. For the EPYC 9005 microarchitecture, this is evident, for example, from AMD documents [**370**, **371**]. In Zen 5, the main development priorities were increasing performance and energy efficiency per core. Information on improvements in Zen 5 cores is available in [**372**, **373**].

Additional data[48] to [**373**] regarding AMD's Hot Chips 2024 keynote includes the relevant slides. Detailed quantitative indicators of both the front-end and back-end components of the Zen 5 core microarchitecture are already presented in [**374**]. It is clear that the progress was integrally expressed in the growth of IPC (by 16%).

We will limit ourselves further to only brief information about the newly released Xeon 6900P and EPYC 9005 models, and some initial data about their performance.

Both companies, in addition to taking advantage of achievements in technology, have moved to work with faster DDR5 memory. AMD processor specifications [**126**] indicate support for up to 12 channels at speeds up to 6000 MT/s, although [**200**] notes support for DDR5-6400 in some models. Xeon 6 processors support up to 12 channels of DDR5-6400 or the higher-performance, albeit more expensive MRDIMM-8800 memory, which has a higher transfer rate. The impact of the increased bandwidth of MRDIMM-8800 on performance compared to using DDR5-6400 is demonstrated in [**375**].

Of course, the big progress in the type of memory supported by Xeon 6 should be reflected in the growth of the achieved bandwidth and, accordingly, in the growth of performance in applications of continuum mechanics, including CFD tasks (this has already become visible in the data of the corresponding `PTS` benchmarks on the openbenchmarking.org website).

[48] `https://chipsandcheese.com/p/discussing-amds-zen-5-at-hot-chips-2024`, accessed 26.04.2025.

The move to higher-bandwidth memory in Xeon 6 appears to be Intel's second most important success after the progress made with the new 5 nm technology. However, the analyses should be conducted not on peak performance valuees, but on the actual performance and memory bandwidth values achieved. For example, the interconnect of the cores indicators in Xeon 6 are important, since potential memory bandwidth limitations were noted for Xeon Max in [**354**].

Intel's STREAM Triad benchmark for a 2P server with 96-core Xeon 6972Ps with MRDIMM-8800 memory showed a 1.3x bandwidth advantage over a 2P server also with 96-core EPYC 9655 and DDR5 memory at a speed of 6000 MT/s [**376**]. In addition, work is already moving forward with Xeon 6 on using CXL memory [**377**].

There is Chinese data attributed to the Chongqing microcomputer[49], for a 2P server with EPYC 9755 on memory bandwidth in the STREAM Triad, as well as on performance in the DGEMM and HPL benchmarks.

It should also be noted here that Xeon 6 was planned to feature a new ISA vector extension, AVX10, with support for a 256-bit vector length limit. And AMD switched to using (when working with AVX-512) a 512-bit data transfer width in EPYC 9005 instead of 256-bit in EPYC 9004. This can not only significantly increase the achieved DGEMM performance compared to EPYC 9004 (which is already visible in the corresponding PTS tests on the openbenchmarking.org website), but also contributes to the growth of performance in AI tasks [**373**]. The impact of this factor on the performance of applications in different areas of use is reviewed in [**378**]. Thus, AMD has moved forward in working with vector operations compared to Zen 4, and Intel in Xeon 6 turned out to be ahead of Zen 5 in the memory technology used.

Intel, thanks to new technology, was able to increase the L3 cache capacity, and some Xeon 6 models in Table 44 have a higher L3 cache capacity per core than the EPYC 9005. AMD, thanks to more advanced semiconductor technology, provides higher clock rates with the same number of cores.

A large amount of information about the EPYC 9005 processors, including performance data in various applications, has already become available on AMD's documentation hub [**146**] among documents that appeared in late 2024. For example, for OpenFOAM performance,

[49] https://news.qq.com/rain/a/20241202A090KQ00, accessed 6.03.2025.

see [**379**]. Relative performance data for the EPYC 9005 compared to the EPYC 9004 and Xeon EMR is provided by AMD for four ANSYS CFD applications[50]. Intel posts unstructured performance data for the Xeon 6 on its website [**377**].

Performance data for EPYC 9005 and Xeon 6 in SPECcpu 2017 benchmarks has also become available. However, the meaning of these benchmarks for modern multi-core server processors has been dramatically reduced. We further rely on the maximum results achieved for the models in question and concrete benchmarks on the official website as of 16.04.2025.

In SPECint, the 128-core Xeon 6980P lags behind the EPYC 9755 with the same core count, while the 96-core Xeon 6972P lags behind the EPYC 9655 with the same core count. In SPECfp, the result is rather the opposite. In SPECfp_rate, which measures the "throughput" of the set of tasks and, unlike SPECfp_speed, parallelization is disabled, the 128-core Xeon 6980P achieved 1360/2720 for 1P/2P configurations versus analogical lower results of 1230/2660 for the EPYC 9755 with the same core count. In SPECfp_speed, the Xeon 6980P achieved 448/496 versus 435/573 for the EPYC 9755 for 1P/2P configurations.

For the 96-core Xeon 6972P, in the SPECfp_rate benchmark achieved 1160/2290 versus 1020/2050 for the EPYC 9655 with the same number of cores in the 1P/2P configuration. In SPECfp_speed, the 1P Xeon 6972P server achieved a maximum of 394 versus 412 for the 1P server with the EPYC 9655.

It is clear that the performance of the Xeon 6900P and EPYC 9005 with the same number of cores is quite close. These results will not be discussed here. We can only note two points. First, there is a big performance improvement of the EPYC 9005 compared to the EPYC 9004 (comparing the 96-core EPYC 9655 and EPYC 9654 models, see data in Table 27); for the Xeon 6900P, the improvement is even greater. Second, we can say that the Xeon 6900P has eliminated the gap with EPYC in the SPECcpu benchmarks.

In the SPECmpi 2007 (medium) benchmark, the 2P server with EPYC 9755 achieved 96.6 versus 65.4 for the 2P server with EPYC 9654 (maximum values as of 14.04.2025), demonstrating a significantly greater performance gain relative to the increase in core count. For the SPEChpc 2021 benchmark, performance data is provided in Table 45. There is no data for Xeon 6 in these SPEC benchmarks.

[50] https://www.amd.com/content/dam/amd/en/documents/epyc-technical-docs/performance-briefs/amd-epyc-9005-pb-ansys-ls-dyna.pdf, accessed 26.04.2025 and similar URLs.

TABLE 45. EPYC 9005 performance in SPEChpc 2021 benchmarks

Model	Number/type of cores	Configuration	Benchmark size: tiny	Benchmark size: small
EPYC 9965	192/Zen 5c	2P	22.9	2.21
EPYC 9755	128/Zen 5	1P	9.24	
		2P	22.1	
EPYC 9555	64/Zen 5	2P	17.7	1.71
EPYC 9754	128/Zen 4c	1P	7.32	
		2P	16.4	
EPYC 9684X	96/Zen 4	2P	16.0	1.55
EPYC 9654	96/Zen 4	1P	6.99	0.735
		2P	14.2	1.59

The table shows only the maximum results achieved as of 14.04.2025.

Unfortunately, the data here with the same number of cores refers to different types of cores.

A lot of performance data for the EPYC 9005 and Xeon 6980P has appeared in PTS benchmarks on openbenchmarking.org. On 15.04.2025, in PTS benchmarks for molecular dynamics (GROMACS, NAMD, LAMMPS), the Xeon 6980P was often only slightly behind the high-end EPYC 9005 models in performance. The lag was also observed in molecular docking (miniBUDE) and quantum chemistry (NWChem) applications.

In some PTS benchmarks related to NPB, the Xeon 6980P was slightly faster, in others, the EPYC 9755. In these PTS benchmark data, special attention must be paid to the "efficiency of scaling" of performance with the number of cores used, which has been discussed many times above.

In terms of AI inference performance, we point out PTS benchmarks based on Intel's OpenVINO tools, in which (as of 17.01.2025) Xeon 6980P processors sometimes outperformed EPYC 9755 and sometimes lagged behind. The author has no data on the use of AMX or optimized Intel software tools in these benchmarks. We will further limit ourselves to comparative data from [**200**].

Table 46 shows a very small number of selected data from the results obtained in [**200**]. The selection was focused on the HPC area, although the table also contains data for the TensorFlow benchmark. The selection made cannot compensate for the above-mentioned shortcomings of some PTS benchmarks, although we tried not to present less informative results.

Table 46. Performance data for Xeon 6, EPYC 9005, and previous generations of Xeon and EPYC in various benchmarks

Model/ Number of cores	n	HPCG 3.1, GFLOPS	NPB 3.4 MG.C, MOPS[2]	Open FOAM medium mesh size, seconds	NWChem, seconds	GROMACS, ns/day	TensorFlow ResNet-50, BS=512, images/s
Xeon 6980P/ 128 (with MRDIMM)[1]	1	86.32	240025.24	123.73	1562.7	20.545	234.72
	2	170.16	451798.00	73.45	1609.5	32.669	180.57
Xeon 6980P/ 128	1	65.22	196892.69	131.37	1564.2	21.086	223.12
	2	127.62	387445.06	71.30	1612.6	33.120	178.53
Xeon 8592+/ 64	1	35.14	107321.89	302.06	1878.3	10.477	140.53
	2	69.30	220893.78	137.04	1809.7	18.540	169.75
Xeon Max 9480/56	1	30.59	86407.46	410.29		8.282	125.14
	2	62.64	171436.47	186.33		14.127	138.39
Xeon 8490H/ 60	1	31.07	83448.70	420.65		8.762	148.45
	2	60.85	163333.21	196.50		18.540	174.95
EPYC 9965/ 192	1	60.70	112588.99	176.40	1557.9	23.142	247.22
	2	93.62	223715.54	79.38	1594.0	30.707	204.43
EPYC 9755/ 128	1	63.67	167026.93	160.37	1326.2	22.729	246.24
	2	64.02	361767.36	74.46	1310.1	34.382	210.75
EPYC 9575F/ 64	1	63.90	159641.78	233.12	1225.7	14.651	254.02
	2	101.90	301867.19	103.87	1127.7	26.665	279.74
EPYC 9754/ 128	1	23.63	125567.07	449.47	1700.0	13.242	135.17
	2	41.87	245447.72	137.11	1712.6	21.305	174.93
EPYC 9684X/ 96	1	27.58	134994.14	178.25	1450.5	15.222	121.96
	2	44.95	244738.81	93.57	1429.9	22.525	137.50
EPYC 9654/ 96	1	28.32	116725.04	325.42	1575.5	12.514	161.90
	2	42.52	194942.32	137.11	1558.4	20.234	133.87

Note: The highest performing benchmark results (separately for 1P or 2P configurations) are marked in red, which also shows whether higher or lower values correspond to higher performance. The data for processors on Zen 5c or Zen 4c cores are marked in blue. All benchmark variants and benchmark results in this table are from the corresponding PTS benchmark results on the openbenchmarking.org website and from [200]. The values used in the table are: n — number of processors, BS — batch size.

[1] In the absence of an indication of the memory used in Xeon 6, it is assumed that it works with DDR5-6400

[2] NPB MG.C was chosen from the NPB 1.4 family of benchmarks by consideration of its greater distance from the HPCG area. In the SP.C and EP.D benchmarks, Xeon 6 was faster, in the CG.C, LU.C and IS.D benchmarks, EPYC 9005 was faster.

This table presents the results [200] for the HPCG (with submesh sizes 144) and OpenFOAM (driverFastback, Medium Mesh Size) benchmarks, where memory bandwidth must be significant to maximize performance, NPB MG.C, and two other HPC applications from the computational

chemistry area that are usually considered rather computationally intensive: NWChem (quantum chemistry) and GROMACS (molecular dynamics).

In addition, the table presents data from [**200**] for a well-known PTS benchmark for AI inference tasks in the RESnet-50 model for the TensorFlow framework for batches of 512. It should be noted here that the performance of such a benchmark scales poorly with an increase in the number of cores when replacing CPU models, when moving from 1P to 2P configurations with EPYC 9004 and Xeon EMR/SPR. And on the latest Xeon 6 and EPYC Zen 5 processors, the performance at all decreased when moving from 1P to 2P configuration. These data are also presented in the table to illustrate that for widely used benchmarks, careful analysis is often required to obtain correct conclusions.

Despite the very short period since the emergence of Xeon 6 and EPYC 9005, the data above even so provides a fairly broad coverage of application areas for the very intial level of evaluation.

Although some of the benchmarks in Table 46 show the performance advantage of the EPYC 9005 and some of the benchmarks show the performance advantage of the Xeon 6, this should not be taken as general. There is already a huge amount of other data in [**200**] or in the PTS benchmarks results on openbenchmarking.org. It is only important to carefully look at what meaning these results have.

But in general, unlike the EPYC 9004 and Xeon SPR/EMR compared in the review, Xeon 6 processors have demonstrated, including in the HPC area, a significantly greater number of performance advantages relative to EPYC 9005 since their appearance than was the case in the area of "around the fourth generation" of x86 server processors.

The statement in [**200**] about the EPYC 9005's performance dominance can only be attributed to the benchmark suite used there; in general, such a statement seems premature. It should also be kept in mind that higher performance could have been achieved there on processors with fewer cores.

From what was said above about performance, it becomes even more important to take into account cost and power consumption. The prices for Xeon 6900P and EPYC 9005 at the time of writing are given in Table 44. It should be noted that earlier the prices for Xeon 6900P were offered by the developer to be significantly higher, significantly exceeding the prices

of analogs (primarily in terms of the number of cores) in EPYC 9005, and in 2025 Intel lowered the prices [**367**]. Although today's prices for similar Xeon and EPYC processors are close, for Xeon they are slightly lower. The specified TDP for EPYC 9755 and Xeon 6980P are the same. But according to [**200**], in the benchmarks used there, the average and peak values of the power used by the EPYC 9965, 9755 and 9575F processors were noticeably lower than those of the Xeon 6980P, which had a peak power consumption of 547 W, noticeably higher than its TDP.

In general, when it comes to comparing the potential efficiency of using Xeon 6 and AMD EPYC Turin, although a lot of data has already appeared on the performance of computing systems based on them, there is still not enough information for a balanced analysis.

7. On Building Nodes of Clusters and Supercomputers using Server x86 Processors

In this section, taking into account the above analysis, some general approaches for using x86 server processors in building clusters and supercomputers for HPC and AI tasks are formulated and references to relevant publications are provided.

For such computing systems today there are 3 different basic variants.

(*1*) Building based on homogeneous nodes

(*2*) Hybrid variants, where one subcluster is built from homogeneous nodes, and the other contains a GPUs (an example of this is the LUMI supercomputer[51])

(*3*) Building using heterogeneous nodes containing x86 processors and GPUs

The optimal choice of processors for variants (*1*) or (*3*), as well as for the corresponding parts of the clusters for the second variant, may be different.

By default, this section assumes that performance is the focus. In the case of orientation to price, price/performance ratio, energy efficiency, or packaging density, this is explicitly stated.

[51] https://docs.lumi-supercomputer.eu/hardware/, accessed 18.04.2025.

Clusters and supercomputers with homogeneous nodes. A classic example of the use of clusters woth homogeneous nodes, focused on big data processing tasks (including working with databases) and AI, is the use of Apache Spark and Hadoop clusters (see, for example, [380–385]). However, you can also use GPU to work with them (see, for example,[387], [387–389]). Although the expediency (taking into account the cost indicators) of using GPUs here in some cases naturally raises questions [390].

Further, it is mainly assumed that the computing systems are focused on HPC. In the case of a cluster of homogeneous nodes, naturally, the optimal x86 processors are those that provide the maximum performance of the node there, where it will be used. For a cluster or supercomputer focused on use in certain pre-known application areas, it may be advantageous, for example, to abandon the need to support the high-performance implementation of AVX-512 provided by scalable 4th and 5th generation Xeon processors, which accordingly provides potential advantages of EPYC Zen 4.

In the case of small clusters, the cost of processors is probably more important. Accordingly, it may be more natural to focus on processors with a higher performance/price ratio, which may be better for mid-range processors [100]. A useful general orientation for building cost-effective clusters with nodes based on the processors considered in this review for processing big data is given in [391]. And for CFD problems, there is, for example, a special methodology for selecting hardware taking into account cost and performance [392].

But we must keep in mind the possibilities of efficient parallelization of the applications used within each node, which is important first of all for such clusters where the area of their application is quite clearly defined. With a very large number of cores in the processor, a situation may arise when, due to the limited memory bandwidth, parallelization using more than a certain number of processor cores may become meaningless, since the performance then practically does not grow or begins to decrease. This may occur with memory-bound applications if the memory bandwidth is saturated even before all the cores available in the processor are used. Of course, the occurrence of a situation with the discommodity of increasing the number of cores in processors depends on the benchmark or application used and the size of the task.

Poor performance scaling with increasing number of cores used in a server is more common and is best known in the field of CFD. For example, in [**393**] the performance of using Runge-Kutta relaxation schemes in a 2P server with 16-core Xeon E5-2698v3 was studied, and the deterioration in performance growth depended weakly on the problem size. Poor performance scaling was revealed still during the period of using multi-core Intel Xeon with MIC (Many Integrated Core) architecture (see, for example, [**394**]).

Naturally, for CFD there is opposite data on the increase of performance when switching to work with a processor containing a larger number of cores and a more expensive one; for EPYC 9004, see, for example, the data from the OpenFOAM and WRF PTS benchmarks in [**147**].

But with an increase in the number of processor cores after a certain threshold the performance of a CFD application can be growing significantly below the linear speedup. In the same benchmarks in [**147**] for the high-end EPYC 9004 models, the performance gain in WRF was very small, and in terms of performance/price, the 64-core EPYC 9554 outperformed the 96-core EPYC 9654.

It is clear that the degradation of performance scaling is only sometimes due to memory bandwidth limitations, although in such cases to work with memory is always given attention (see, for example, [**395**]).

Similar situations arise when working with DGEMM. Thus, in [**396**] for DGEMM from MKL, the performance growth of a 2P server with 24-core Xeon 8160 (Skylake) stopped when using 16 cores, and then the performance dropped. In [**397**], the performance of DGEMM from the Intel-developed Parallel Research Kernels in a 2P server with 8-core Xeon E5-2450 stopped growing at 8 cores.

Similar situations in clusters are more complex. A special term, undersubscription, has also appeared, which is used in cases where it makes sense to use fewer cores than are available at the calculation in cluster nodes to achieve higher performance, and the reason for this has not yet been identified. Naturally, this is observed in CFD. Thus, it was shown that in the Ansys Fluent application, undersubscription when working with 60-core EPYC Zen 3 (7V73X) can speed up the calculation by up to 50% [**395**]. It is possible that the undersubscription effect occurs in the case of parallelization using turbo frequencies of processors in the cluster nodes.

Using processors with fewer cores in nodes not only results in a significantly lower cost, which is usually more important for a cluster, but also possibly a certain increase in performance due to higher clock frequencies in such processors, which is typical for EPYC 9004 models. Similarly, it may be more effective, for example, to use Xeon Gold in nodes rather than Xeon Platinum. This corresponds to what was said above about the possible focus on mid-range processor models. In the extreme case, when the need for the achieved performance level is not so high, it may turn out that in terms of price/performance ratio it is more profitable to create a cluster of two nodes with processors with fewer cores than to use one server with processors with a larger number of cores.

On the other hand, with an increase in the number of nodes in the cluster, a deviation of the performance scaling to the side significantly worse than its linear growth can also be observed. To determine how many nodes in the cluster and which nodes/processors are optimal to choose for solving CFD problems, a general methodology is proposed in [**398**]. It takes into account performance and cost (and therefore energy efficiency). Although this methodology was demonstrated using the Fluent application as an example, it is applicable to other applications as well.

For large clusters and supercomputers, their clear focus on specific applications is unlikely; rather, it is assumed that a wide variety of unpredictable applications will be used. Accordingly, optimizing the choice of processors for concrete applications is meaningless. For supercomputers, top models of the corresponding processor families are often selected, and energy efficiency (and, accordingly, TDP) and packaging density are more important. The relevant issues of practical and efficient construction of supercomputer nodes with a performance of over 100 TFLOPS (including a focus on exascale supercomputers and the use of GPUs) are considered in [**399**].

Clusters and supercomputers with heterogeneous nodes. For cases of building a computing system with heterogeneous nodes containing x86 processors and GPUs, the situation is mostly different.

In heterogeneous nodes with GPUs, the requirements for processors are much different than for homogeneous nodes. The tasks of x86 processors are to load and service the OS, run programs, and most importantly, to exchange data with one or more GPUs in the node. In the "limit case", almost all performance provision is transferred to the GPU, the processor's task is mainly communication with the GPU (or — not necessarily — between nodes). And all calculations are essentially performed on the GPU. Then the main thing is to ensure fast data exchange with the GPU, and not the high performance of the processors themselves.

While all supercomputers require maximizing energy efficiency and packing density, this may become even more important for GPU-containing nodes (and GPUs improve energy efficiency). Therefore, ARM processors, which are considered primarily energy efficient, become very relevant here.

First, we will consider the variant where the processors and GPUs in the nodes are separated from each other, and then the variant with processors and GPUs integrated into a single chip.

As noted in [**391**], when using RDMA, the use of IOD in EPYC 9004 with its integrated memory and PCIe controller provides a short path between memory and PCIe, which helps reduce latency and increase the bandwidth utilization of data exchanges between memory and PCIe. This is important for computing systems containing GPUs, which are often connected via PCIe. In Xeon EMR, however, memory and PCIe access is distributed among the cores, which increases the path length of such data transfers. However, in other situations, this is a plus for Xeon EMR [**391**]. It should be noted here that Xeon 6 also began to use IOD (see Section 6 above).

This is becoming more important as modern supercomputers increase the number of GPUs in node per processor [**144**]. For example, Titan had a 1:1 ratio, Summit had a 3:1 ratio, and Frontier had a 4:1 ratio [**399**]. And Frontier now has over 99% of its floating point operations executed by GPUs [**144**].

Accordingly, maximizing the bandwidth of the RAM with the processor becomes more important, which requires using an optimal NUMA configuration. For example, in 1P nodes of the Frontier supercomputer with 64-core EPYC 7A53, NPS=4 is used. The bandwidth in STREAM benchmarks with NPS=4 reached 180 GB/s, and with NPS=1 — only about 125 GB/s [144].

Also more important becomes support in the processor of a possible specific GPU interconnect of the manufacturer, or universal PCIe or CLX. The use of ARM processors for these purposes may be more effective, which was implemented in the integrated Nvidia GH200 chip [2].

But today, the vast majority of servers with GPUs use x86 Xeon or EPYC processors. Perhaps the reduced performance requirements of multi-core x86 processors in heterogeneous nodes, even containing multiple GPUs, lead to the fact that in servers-nodes of supercomputers it is often sufficient to use a single processor with a large number of cores, rather than working with traditional for HPC 2P servers.

An example of this is the powerful supercomputers of the November 2024 TOP500 list, which use one optimized 64-core EPYC Trento Zen 3 with four AMD MI250X GPUs in their nodes. This is the former leader of the list, Frontier [144], the HPC6 supercomputer ranked 5th on this list (Europe's leading in performance) [400], and ranked 8th LUMI [401]. And Perlmutter [402], which is in 19th place, contains one 56-core EPYC 7763 with four Nvidia A100 GPUs in its nodes. These supercomputers use modern GPUs from both AMD and Nvidia. But one processor with fewer cores in a node may not be enough.

One can also point to the German HoreKa-Teal supercomputer from the TOP500, where in nodes, together with four Nividia H100 GPUs, two 32-core EPYC 9354 are used [145].

Examples with AMD EPYC processors were given here, since they are actively used in the leading TOP500 supercomputers together with both AMD and Nvidia GPUs. This is largely due to the active use of ready HPE Cray nodes in supercomputers, for example, EX235a (with AMD GPU) and EX235n (with Nvidia GPU) [403]. The brief description of these nodes in [403] only indicates the number of sockets and the type of processor, but not its concrete model.

Xeon 8480C processors, containing 56 cores, are also often used in supercomputer nodes with Nvidia H100 GPUs. And in the Aurora supercomputer (second place in the TOP500 list), nodes use two 52-core Xeon Max 9470C and 6 GPUs from Intel (previously called Ponte Vecchio) [354]. To clarify, we note that models with the C suffix differ from the "regular" ones only in slightly lower maximum temperature indicators and the absence of some options that are not interesting for the review tasks.

Here it is also necessary to pay attention to the fact that an increasing percentage of servers with GPUs or heterogeneous nodes containing GPUs are used for AI tasks. And there, for the largest-scale such tasks, a huge memory capacity is required, and in the "GPU part" as well. But accordingly, the server/node processors must be able to work with a very large address space and in the physical plane as well, and the expected active use of CXL memory, which is supported by the AMD and Intel processors considered in the review, can become a big potential plus.

The use in nodes of processors integrated with GPUs removes the issue of selecting processors for the node, since this has already been done by the manufacturer. A number of supercomputers in the November 2025 TOP500 list use Nvidia GH200 in nodes, where the 72-core ARM Grace [13, 357] is integrated with the GPU (the highest among them, the seventh place is occupied by the Alps supercomputer; see about it in [404]). But AMD MI300A, which has a 24-core Zen 4 processor integrated with the GPU, was no less actively used in this list [405]. The TOP500 leader, the exascale supercomputer El Capitan of the Lawrence Livermore National Laboratory, contains four MI300A in nodes. These supercomputers also use ready HPE Cray EX254n and EX255a nodes [403].

Conclusion

The following conclusions apply to the conditional fourth generation of Xeon and EPYC considered in the review, unless otherwise explicitly stated (for example, for Xeon 6 or EPYC Zen 5 processors).

1. The current state of x86-64 server processors shows that, in general, the advantage in performance and energy efficiency (as well as in terms of cost) today goes to the company that has an advantage in the semiconductor technology used (AMD with TSMC technology therefore often outpaced Intel). This is confirmed by Intel's success with Xeon 6 after the appearance of the new Intel 5 nm technology.
2. In the case of using modern semiconductor technologies from 10 nm (in Intel Xeon SPR) and below, the optimal solution for such processors is to use chiplets and, possibly, three-dimensional technologies.
3. Modern ARM processors continue to lag behind high-end x86 processor models in terms of performance (and performance per core), which are referred to in the review as the conditional 4th generation. Although energy efficiency is usually considered an advantage of ARM processors over x86-64 server processors, the SPECpower_ssj 2008 benchmark data showed the opposite results.
4. While EPYC 9004 processors outperform Xeon SPR (including Xeon Max) and Xeon EMR in most common server application areas, there are many important, narrower niches where these Intel Xeon generations can perform better.

   In the HPC area, these could be Xeon EMR with applications where performance is limited by matrix multiplication. Although this applies to a fairly small part of HPC, and the author does not know of any direct data on higher DGEMM performance for Xeon 8592+ compared to high-end EPYC 9004 models. The larger number of cores in such EPYC models eliminates the possible advantages of Xeon EMR. And, for example, in the HPL benchmark, which actively uses matrix multiplication, there is the above data on higher EPYC performance.

   In the AI area (unless this task is better suited to a GPU), Xeon (especially Xeon EMR) can be more efficient due to the use of the AMX extension in ISA.

   In the in-memory DB area, Xeon's advantage is due to its support for up to 8 sockets in servers, which allows it to work with very large shared memory capacities that are not available for 2P servers. Support for working with Xeon-based servers with more than the "normal" two sockets is a general advantage that can manifest itself in other areas of application, such as AI tasks.

   The use of specialized accelerators in Xeon creates other opportunities for the formation of such niches.

5. Although the advantages of AMD server processors began to emerge quite a long time ago (after the appearance of AMD Zen 2 Rome processors in 2019), with EPYC 9004, the area of their optimal application (up until the appearance of Intel Xeon 6 Granite Rapids) expanded to the maximum.

   With the introduction of Intel Xeon 6 Granite Rapids (Xeon 6900P) and AMD Zen 5 Turin (EPYC 9005) processors with new semiconductor technologies in the fall of 2024, situation will change. Although the EPYC 9005 processors show higher performance in many cases, the Xeon 6900P demonstrate possible liquidation of quite long-term lag of Intel server processors by performance. With the Xeon 6900P, Intel also made success due to increased memory bandwidth (where it had previously lagged behind AMD in memory-bound applications), and AMD has made some progress in working with AVX-512, where it could lag in performance.

6. As the number of cores in processors has grown, situations have become more common where, for traditional HPC applications, using high-end and more expensive models with a very large number of cores in concrete calculations can give lower performance than using processors with fewer cores (and with higher clock rates). Previously, this was known mainly for CFD tasks, when it was more efficient to use clusters with processors with fewer cores.

   High-end models with more cores may have lower theoretical memory bandwidth per core, and available data, for example, for Xeon SPR and Xeon Max, indicate a significant decrease in memory bandwidth gains with increasing core counts. All of this contributes to worse parallelization with increasing core counts.

   This has led to a some paucity of data from widely performed benchmarks (including those using applications), as they usually do not analyze the performance dependence on the number of cores used (for example, in the PTS benchmarks at openbenchmarking.org). As a result, some performance estimates in many ways diminish their meaning and do not demonstrate the effectiveness of using different processor models.

7. The use of hardware security support in the processors in question may expand as the use of cloud technologies expands. The new TDX technology created by Intel was based on a huge amount of work, including collaboration with other well-known companies. TDX includes extensions to ISA and requires application modernization to use it. Therefore, TDX is a great reserve for the future, primarily for cloud applications. AMD with EPYC uses hardware-only tools for security, which is simpler in essence.

   However, data that security features in modern processors cause large performance hits for HPC, and the need to upgrade application code to work with TDX, make this ineffective for HPC.

8. Intel's most important potential advantage remains its software development tools, including oneAPI with DPC++. Despite the observed performance advantages of modern AMD server processors, traditionally widespread use of Intel's SDKs may help with orientation on Xeon. Although the use of oneAPI as an open standard with open source code may partly offset the advantages of Xeon over EPYC.

9. It is possible that the use of modern server processors for AI tasks will expand, including with the use of Apache Hadoop or Spark clusters. First of all, this is likely in relation to AI inference. But in what cases and how this can really effective as compared with working on a GPU is not yet entirely clear.

10. As for supercomputer technologies and cluster building, the data discussed above previously gave advantages for EPYC when used both in homogeneous nodes and in heterogeneous nodes using GPUs. For homogeneous nodes without GPUs, the emergence of Xeon 6 provides a possible leveling of the performance gap in this generation of Xeon compared to Zen 5. In heterogeneous nodes with GPUs, there may be slightly different tasks for processors, with greater requirements for memory bandwidth, which may currently be higher in Xeon 6 with higher-speed memory. AMD's primary focus on HPC tasks may also be important, although EPYC 9004 is also actively used for AI tasks.

References

[1] *Top500 the list*, The Top500 ranking, 63rd, 2025. URL ↑44, 54

[2] M. B. Kuz'minskij. "New generation of GPGPU and related hardware: computing systems microarchitecture and performance from servers to supercomputers", *Program Systems: Theory and Applications*, **15**:2(61) (2024), pp. 139–473 (in Russian). URL doi ↑44, 45, 49, 65, 73, 81, 168, 244

[3] B. Drávai, I. Z. Reguly. "Benchmarking the evolution of performance and energy efficiency across recent generations of Intel Xeon processors", *PMBS 2024: 15th IEEE International Workshop on Performance Modeling, Benchmarking, and Simulation of High Performance Computer Systems* (Atlanta, GA, USA, 17–22 November 2024), IEEE, 2024, ISBN 979-8-3503-5554-3, pp. 1413–1419. doi ↑44, 217, 226

[4] L. A. Torres, C. J. Barrios, Y. Denneulin. *Evaluation of computational and energy performance in matrix multiplication algorithms on CPU and GPU using MKL, cuBLAS and SYCL*, 2024, 14 pp. arXiv 2405.17322 ↑44, 190

[5] *Top500 the list*, The Top500 ranking, 61sd, 2023. URL ↑44

[6] X. Duan, J. Wang, P. Gao, M. Ma, L. Gan, X. Liu, H. Fu, W. Xue, D. Chen, G. Yang, W. Liu. "Enabling real world scale structural superlubricity all-atom simulation on the next-generation sunway supercomputer", *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Denver, CO, USA, 12–17 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0109-2, id. 99, 14 pp. doi ↑44

[7] M. B. Kuz'minskij. "Sovremennye servernye ARM-processory dlya superEVM: A64FX i drugie. Nachal'nye dannye testov proizvoditel'nosti", *Program Systems: Theory and Applications*, **13**:1 (52) (2022), pp. 63–129; M. B. Kuzminsky. "Modern server ARM processors for supercomputers: A64FX and others. Initial data of benchmarks", *Program Systems: Theory and Applications*, **13**:1(52) (2022), pp. 131–194. URL doi URL doi ↑44, 70, 168

[8] N. A. Simakov, R. L. Deleon, J. P. White, M. D. Jones, T. R. Furlani, E. Siegmann, R. J. Harrison. "Are we ready for broader adoption of ARM in the HPC community: Performance and energy efficiency analysis of benchmarks and applications executed on high-end ARM systems", *International Conference on High Performance Computing in Asia-Pacific Region Workshops*, HPCASIAWORKSHOP 2023 (Raffles Blvd, Singapore, February 27–March 2, 2023), ACM, New York, 2023, ISBN 978-1-4503-9989-0, pp. 78–86. doi ↑44, 52, 53

[9] T. N. Rahman, N. Khan, Z. I. Zaman. *Redefining computing: Rise of ARM from consumer to Cloud for energy efficiency*, 2024, 19 pp. arXiv 2402.02527 ↑44

[10] D. Loghin. "Are ARM cloud servers ready for database workloads? An experimental study", *IEEE Transactions on Cloud Computing*, **12**:3 (2024), pp. 818–829. doi ↑44

[11] R. Matha, D. Kimovski, A. Zabrovskiy, C. Timmerer, R. Prodan. "Where to encode: A performance analysis of x86 and arm-based Amazon EC2 instances", *2021 IEEE 17th International Conference on eScience (eScience)* (Innsbruck, Austria, 20–23 September 2021), IEEE, 2021, ISBN 978-1-6654-0361-0, pp. 118–127. ↑44

[12] M. Larrabel. *NVIDIA GH200 CPU Performance Benchmarks Against AMD EPYC™ Zen 4 & Intel Xeon Emerald Rapids*, Processors Linux Reviews & Articles, Phoronix Media, 2024, 5 pp. ↑45, 112

[13] J. Laukemann, G. Hager, G. Wellein. *Microarchitectural comparison and in-core modeling of state-of-the-art CPUs: Grace, Sapphire Rapids, and Genoa*, 2024, 5 pp. arXiv 2409.08108 ↑45, 150, 151, 245

[14] E. Siegmann, R. J. Harrison, D. Carlson, S. Chheda, A. Curtis, F. Coskun, R. Gonzalez, D. Wood, N. A. Simakov. "First impressions of the sapphire rapids processor with HBM for scientific workloads", *SN Computer Science*, **5**:5 (2024), id. 623, 11 pp. ↑45, 172, 219, 223, 224

[15] J. B. Kotra, J. Kalamatianos. "Improving the utilization of micro-operation caches in x86 processors", *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)* (Athens, Greece, 17–21 October 2020), IEEE, 2020, ISBN 978-1-7281-7384-9, pp. 160–172. ↑45

[16] X. Ren, L. Moody, M. Taram, M. Jordan, D. M. Tullsen, A. Venkat. "I see dead μops: Leaking secrets via Intel/AMD micro-op caches", *ISCA '21: Proceedings of the 48th Annual International Symposium on Computer Architecture* (Virtual Event, Spain, 14–19 June 2021), IEEE, 2021, ISBN 978-1-4503-9086-6, pp. 361–374. ↑45

[17] *Introducing Intel Advanced Performance Extensions (Intel APX) Upd. 3/5/2024*, 2024 (Accessed 31.07.2024$^{(*)}$). ↑45

[18] E. Blem, J. Menon, K. Sankaralingam. *A detailed analysis of contemporary ARM and x86 architectures*, UW-Madison Technical Report, 2013 (Accessed 12.05.2025), 13 pp. ↑45

[19] E. Blem, J. Menon, K. Sankaralingam. "Power struggles: Revisiting the RISC vs. CISC debate on contemporary ARM and x86 architectures", *HPCA '13: Proceedings of the 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)* (23–27 February 2013), IEEE, 2013, ISBN 978-1-4673-5585-8, pp. 1–12. ↑45

[20] E. Blem, J. Menon, T. Vijayaraghavan, K. Sankaralingam. "ISA wars: Understanding the relevance of ISA being RISC or CISC to performance, power, and energy on modern architectures", *ACM Transactions on Computer Systems (TOCS)*, **33**:1 (2015), id. 3, 34 pp. ↑45

[21] Ch. Lam. *Why x86 doesn't need to die*, 2024 (Accessed 12.05.2025). ↑46

[22] K. Troester, R. Bhargava. "AMD next generation "Zen 4" Core and 4th Gen AMD EPYC™ 9004 server CPU", *2023 IEEE Hot Chips 35 Symposium (HCS)* (Palo Alto, CA, USA, 27–29 August 2023), IEEE, 2023, ISBN 9798350339086, pp. 1–25. ↑46, 66, 67, 68, 70, 71, 72, 74, 76, 80

[23] *4th Gen AMD EPYC™ processor architecture*, White Paper 3rd Ed, 2023 (Accessed 2.08.2024). URL ↑46, 58, 68, 70, 71, 73, 76, 77, 79, 81, 83, 84, 88, 89, 142

[24] P. Polgár, T. Menyhárt, B. C. Baksay, G. Kocsis, T. G. Tajti, Z. Gál. "Three level benchmarking of Singularity containers for scientific calculations", *Annales Mathematicae et Informaticae*, **58** (2023), pp. 133–146. doi ↑46

[25] L. Szustak, R. Wyrzykowski, L. Kuczynski, T. Olas. "Architectural adaptation and performance-energy optimization for CFD application on AMD EPYC™ Rome", *IEEE Transactions on Parallel and Distributed Systems*, vol. **32**, 2021, pp. 2852–2866. doi ↑46

[26] *Cloud/CPU/Java/Power/Storage/Virtualization Benchmarks* (Accessed 19.12.2024). URL ↑47

[27] *Processors benchmarks results* (Accessed 19.12.2024). URL ↑47

[28] *Official Phoronix Test Suite tests* (Accessed 19.12.2024). URL ↑47

[29] M. Larabel. *AMD EPYC™ 7302/7402/7502/7742 Linux Performance Benchmarks*, 2019 (Accessed 12.05.2025), 9 pp. URL ↑47

[30] M. Larabel. *AMD EPYC™ 7003 "Milan" Linux Benchmarks - Superb Performance*, 2021 (Accessed 12.05.2025), 12 pp. URL ↑47

[31] M. Larabel. *Intel Xeon Ice Lake vs. AMD EPYC™ Milan Server performance, efficiency & value in*, 2023 (Accessed 12.05.2025), 10 pp. URL ↑47

[32] S. Akter, K. Khalil, M. Bayoumi. "A survey on hardware security: Current trends and challenges", *IEEE Access*, **11** (2023), pp. 77543–77565. doi ↑47

[33] A. Francillon, C. Castelluccia. "Code injection attacks on harvard-architecture devices", *CCS '08: Proceedings of the 15th ACM conference on Computer and communications security* (Alexandria, Virginia, USA, 27–31 October 2008), ACM, New York, 2008, ISBN 978-1-59593-810-7, pp. 15–26. doi ↑47

[34] J. Zhang, C. Chen, J. Cui, K. Li. "Timing side-channel attacks and countermeasures in CPU microarchitectures", *ACM Computing Surveys*, **56**:7 (2024), id. 178, 40 pp. doi ↑47

[35] W. Chen, Yu. Zhao, Y. Zhang, W. Qiang, D. Zou, H. Jin. "ReminISCence: trusted monitoring against privileged preemption side-channel attacks", *Computer Security — ESORICS 2024: 29th European Symposium on Research in Computer Security* (Bydgoszcz, Poland, 16–20 September 2024), Springer-Verlag, Berlin–Heidelberg, 2024, ISBN 978-3-031-70902-9, pp. 24–44. doi ↑47

[36] J. Wikner, K. Razavi. "Breaking the barrier: post-barrier spectre attacks", *2025 IEEE Symposium on Security and Privacy (SP)* (San Francisco, CA, USA, 12–15 May 2025), IEEE, ISBN 979-8-3315-2236-0, pp. 3516–3533. doi ↑47, 175

[37] L. Li, H. Yavarzadeh, D. Tullsen. "Indirector: high-precision branch target injection attacks exploiting the indirect branch predictor", *SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), USENIX Ass., Berkeley, 2024, ISBN 978-1-939133-44-1, pp. 2137–2154, id. 120. doi ↑47, 175

[38] E. Aktas, C. Cohen, J. Eads, J. Forshaw, F. Wilhelm. *Intel Trust Domain Extensions (TDX) security review*, Google technical report, 2023 (Accessed 12.05.2025), 81 pp. URL ↑48, 164, 165

[39] J. Wikner, D. Trujillo, K. Razavi. "Phantom: exploiting decoder-detectable mispredictions", *MICRO '23: Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture* (Toronto, ON, Canada, 28 October 2023–1 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0329-4, pp. 49–61. doi ↑48

[40] D. Trujillo, J. Wikner, K. Razavi. "INCEPTION: exposing new attack surfaces with training in transient execution", *SEC '23: Proceedings of the 32nd USENIX Conference on Security Symposium* (Anaheim, CA, USA, 9–11 August 2023), USENIX Ass., Berkeley, 2023, ISBN 978-1-939133-37-3, pp. 7303–7320, id. 409. doi ↑48

[41] P. L. Wang, R. Paccagnella, R. S. Wahby, F. Brown. "Bending microarchitectural weird machines towards practicality", *SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), USENIX Ass., Berkeley, 2024, ISBN 978-1-939133-44-1, pp. 1099–1116, id. 62. doi ↑48

[42] P. Jattke, M. Wipfli, F. Solt, M. Marazzi, M. Bölcskei, K. Razavi. "ZENHAMMER: Rowhammer attacks on AMD zen-based platforms", *SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), USENIX Ass., Berkeley, 2024, ISBN 978-1-939133-44-1, pp. 1615–1633. doi ↑48

[43] S. Gast, J. Juffinger, L. Maar, C. Royer, A. Kogler, D. Gruss. *Remote scheduler contention attacks*, 2024, 22 pp. arXiv 2404.07042 ↑48

[44] T. Kim, H. Jang, Y. Shin. *Demoting security via exploitation of cache demote operation in Intel's latest ISA extension*, 2025, 18 pp. arXiv 2503.10074 ↑48

[45] *Intel C741 Chipset* (Accessed 28.07.2024(*)). URL ↑49, 166

[46] V. Pano, R. Kuttappa, B. Taskin. "3D NoCs with active interposer for multi-die systems", *Proceedings of the 13th IEEE/ACM International Symposium on Networks-on-Chip* (New York, 17–18 October 2019), ACM, New York, 2019, ISBN 978-1-4503-6700-4, pp. 1–8, id. 14. doi ↑49

[47] *Second Generation Intel Xeon Scalable Processors Product Brief*, 2019 (Accessed 12.05.2025(*)). URL ↑49

[48] B. Atkins, J. Chang, J. Hatch, D. Lee, Ch. Napombejara, M. Popp. *The Future of AMD*, 2007 (Accessed 28.07.2024), 39 pp. URL ↑50

[49] *CPU Specs Database* (Accessed 17.12.2024). URL ↑50, 74, 78, 102, 128, 139, 142, 155, 183, 215, 231, 232

[50] *Intel Processors Product Specifications* (Accessed 23.02.2025(*)). URL ↑50, 155

[51] *All SPEC CPU2017 Results Published by SPEC* (Accessed 19.05.2025). URL ↑50, 66, 214

[52] *High End CPUs – Intel vs AMD*, CPU Benchmarks, 2024 (Accessed 29.07.2024). URL ↑50, 51

[53] J. D. McCalpin. *Memory bandwidth and system balance in HPC systems*, SC16 Invited Talk: Memory Bandwidth and System Balance in HPC Systems, 2016 (Accessed 12.05.2025). URL ↑51

[54] R. H. Katz, J. L. Hennessy. *High performance microprocessor architectures*, 1976 (Accessed 19.05.2024), 17 pp. URL ↑51

[55] C. S. Guiang, K. F. Milfeld, A. Purkayastha, J. R. Boisseau. "Memory performance of dual-processor nodes: comparison of Intel Xeon and AMD Opteron memory subsystem architectures", *4th LCI International Conference on Linux Clusters: The HPC Revolution* (San Jose, California, USA, 23–26 June 2003), 2003 (Accessed 2.05.2025), 24 pp. URL ↑51

[56] J. D. McCalpin. *The evolution of single-core bandwidth in multicore processors*, 2023; *The evolution of single-core bandwidth in multicore systems — update* (Accessed 21.05.2025). URL URL ↑51

[57] M. Guest, J. Munoz, T. Green. "Performance of community codes on multi-core processors. An analysis of computational chemistry and ocean modelling applications", *Computing Insight UK 2022 Conference* (Manchester Central Convention Complex, 1–2 December 2022), 2022 (Accessed 22.11.2024). URLGuest.pdf ↑51

[58] *HPC Challenge Benchmark*, 2022 (Accessed 1.12.2024). URL ↑52

[59] E. Eshelman. *Detailed specifications of the "Ice Lake SP" Intel Xeon Processor Scalable Family CPUs*, 2021 (Accessed 12.05.2025). URL ↑53, 54

[60] E. Eshelman. *Detailed specifications of the AMD EPYC™ "Milan" CPUs*, 2021 (Accessed 12.05.2025). URL ↑53, 55

[61] *Top500 the list*, List Statistics of TOP500, 63rd edition, 2024 (Accessed 12.05.2025). URL ↑54

[62] R. Nana, C. Tadonki, P. Dokladal, Y. Mesri. *Energy concerns with HPC systems and applications*, 2023, 20 pp. arXiv 2309.08615 ↑56

[63] *Fujitsu server PRIMERGY performance report*, PRIMERGY RX8770 M7 Vers.1.1, 2023 (Accessed 31.07.2024). URL ↑56, 190, 193, 198

[64] *Calculate the Max Flops on Skylake*, 2018 (Accessed 8.02.2025). URL ↑56

[65] D. L. Mulnix. *Third generation Intel Xeon processor Scalable family on two socket platform technical overview*, 2024 (Accessed 4.12.2024(*)). URL ↑57

[66] *Intel Xeon 6*, 2024 (Accessed 4.10.2024(*)). URL ↑57

[67] M. P. Karpowicz. "Energy-efficient CPU frequency control for the Linux system", *Concurrency and Computation: Practice and Experience*, **28**:2 (2016), pp. 420–437. doi ↑57

[68] D. Brodowski, N. Golde, R. J. Wysocki, V. Kumar. *CPU frequency and voltage scaling code in the Linux kernel*, 2017 (Accessed 12.05.2025). URL ↑57, 59, 60

[69] A. A. Sinkar, H. Wang, N. S. Kim. "Workload-aware voltage regulator optimization for power efficient multi-core processors", *2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (Dresden, Germany, 12–16 March 2012), IEEE, 2012, pp. 1134–1137. doi ↑57

[70] *Advanced Configuration and Power Interface (ACPI) Specification*, Release 6.5, UEFI Forum, Inc., 2022 (Accessed 12.05.2025), 1126 pp. URL ↑57

[71] J. Chen, R. Wolford. *Understanding P-state control on Intel Xeon Scalable Processors to maximize energy efficiency*, 2019 (Accessed 12.05.2025), 34 pp. URL ↑58

[72] *CPU frequency scaling*, 2025 (Accessed 8.02.2025). URL ↑58

[73] *OpenSuSE Leap 15.5 Power Management* (Accessed 8.02.2025). URL ↑58, 59, 60

[74] *Chapter 17. Tuning CPU frequency to optimize energy consumption*, A Red Hat training course, 2024 (Accessed 8.02.2025). URL ↑58, 59, 60

[75] *oneAPI — What is it?* 2024 (Accessed 11.08.2024$^{(*)}$). URL ↑61, 64

[76] *High Performance Toolchain: Compilers, Libraries & Profilers*, Tuning Guide AMD EPYC™ 9004. Rev. 1.4, 2023 (Accessed 5.10.2024), 54 pp. URL ↑62

[77] *AMD Optimizing C/C++ and Fortran Compilers (AOCC)*, 2024 (Accessed 3.09.2024). URL ↑62

[78] *AMD AOCC User Guide*, 2024 (Accessed 3.09.2024), 42 pp. URL ↑62

[79] M. Larabel. *AMD AOCC 5.0 Compiler Released With Zen 5 Support, new optimizations*, 2024 (Accessed 12.10.2024). URL ↑62

[80] *HPE Cray Compiler Environment*, 2024 (Accessed 12.05.2025). URL ↑62

[81] *HPE Cray Programming Environment User Guide: HPCM on HPE Cray:Supercomputing EX and HPE Cray Supercomputing Systems (24.07)*, S-8023, 2024 (Accessed 3.10.2024), 33 pp. URL ↑62

[82] *HPE Cray Supercomputing Programming Environment Software Overview*, 2024 (Accessed 27.10.2024), 9 pp. URL ↑62

[83] *NVIDIA HPC SDK Version 24.7 Documentation*, 2024 (Accessed 3.09.2024). URL ↑62

[84] O. W. Saastad, K. Kapanova, S. Markov, C. Morales, A. Shamakina, N. Johnson, E. Krishnasamy, S. Varrette. *Best Practice Guide Modern Processors*, ed. Shoukourian H., PRACE, 2020 (Accessed 28.11.2024), 106 pp. URL ↑63

[85] K. Halbiniak, R. Wyrzykowski, L. Szustak, A. Kulawik, N. Meyer, P. Gepner. "Performance exploration of various C/C++ compilers for AMD EPYC™ processors in numerical modeling of solidification", *Advances in Engineering Software*, **166** (2022), id. 103078. doi ↑63

[86] R. R. L. Machado, J. Eitzinger, J. Laukemann, G. Hager, H. Köstler, G. Wellein. *MD-Bench: Engineering the in-core performance of short-range molecular dynamics kernels from state-of-the-art simulation packages*, 2023, 17 pp. arXiv 2302.14660 ↑63

[87] D. Dal, E. Celik. "Investigation of the impact of different versions of GCC on various metaheuristic-based solvers for traveling salesman problem", *The Journal of Supercomputing*, **79**:11 (2023), pp. 12394–12440. doi ↑63

[88] *4th Gen AMD EPYC™ Processor Benchmark Report v. 1.0*, CC00204, 2023 (Accessed 11.10.2024), 22 pp. URL ↑63, 66, 69, 103, 110, 111, 127, 128, 129, 130

[89] *EPYC™ CPU Overview*, 2024 (Accessed 6.10.2024), 46 pp. URL Sinica_HPC workshop.pdf ↑63, 94, 95

[90] J. Wang, E. Day. *LS-DYNA and high-performance computing*, 2023 (Accessed 27.10.2024), 20 pp. URL ↑63

[91] M. Larabel. *LLVM Clang 16 vs. GCC 13 Compiler Performance On AMD 4th Gen EPYC™ "Genoa"*, 2023 (Accessed 20.10.2024), 6 pp. URL ↑63

[92] *AMD Zen Deep Neural Network (ZenDNN)*, 2024 (Accessed 13.10.2024). URL ↑64

[93] *AI Inferencing with AMD EPYC™ Processors*, 2024 (Accessed 09.01.2026), 11 pp. URL ↑64

[94] *IFX vs IFORT performance difference*, 2023 (Accessed 5.09.2024). URL ↑65

[95] *Intel HPC Toolkit Documentation*, 2024 (Accessed 12.05.2025(*)). URL ↑65

[96] *AI Tools Documentation*, 2024 (Accessed 12.05.2025(*)). URL ↑65

[97] *Community building blocks for HPC systems* (Accessed 8.09.2024). URL ↑65

[98] *Intel oneAPI Base Toolkit Documentation*, 2024 (Accessed 8.09.2024(*)). URL ↑66

[99] D. Ruhela. "Investigating the performance of LLVM-Based Intel Fortran Compiler (ifx)", *High Performance Computing*, ISC High Performance 2024 International Workshops. ISC High Performance 2023 (Hamburg, Germany, 12–16 May 2024), Lecture Notes in Computer Science, vol. **15058**, Springer, Cham, 2025, ISBN 978-3-031-73715-2, pp. 173–184. DOI ↑66

[100] M. Cuma. *AMD Genoa and Intel Sapphire Rapids review*, 2023 (Accessed 13.08.2024), 12 pp. URL ↑66, 72, 109, 113, 114, 116, 181, 213, 214, 240

[101] B. Munger, K. Wilcox, J. Sniderman, C. Tung, B. Johnson, R. Schreiber. ""Zen 4": The AMD 5nm 5.7 GHz x86-64 microprocessor core", *2023 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 19–23 February 2023), IEEE, 2023, ISBN 978-1-6654-9017-7, pp. 38–39. DOI ↑67, 68

[102] *Zen 2 — Microarchitectures — AMD* (Accessed 4.10.2024). URL ↑67

[103] *Zen 3 — Microarchitectures — AMD* (Accessed 4.10.2024). URL ↑67, 69, 84

[104] *Zen 4 — Microarchitectures — AMD* (Accessed 4.10.2024). URL ↑67, 68, 69, 84

[105] *AMD "Zen 4" Dies, Transistor-Counts, Cache Sizes and Latencies Detailed*, 2022 (Accessed 12.05.2025). URL ↑69

[106] Ch. Lam. *AMD's Zen 4 Part 1: Frontend and Execution Engine*, 2022 (Accessed 4.10.2024). URL ↑68

[107] R. Bhargava, K. Troester. "AMD Next Generation "Zen 4" core and 4th Gen AMD EPYC™ server CPUs", *IEEE Micro*, **44**:3 (2024), pp. 8–17. DOI ↑68, 76

[108] *AMD Zen 4 13 Percent IPC Uplift Build* (Accessed 21.04.2024). URL ↑69

[109] J. Domke, E. Vatai, A. Drozd, P. ChenT, Y. Oyama, L. Zhang. Matrix engines for high performance computing: A paragon of performance or grasping at straws? *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (Portland, OR, USA, 17–21 May 2021), IEEE, 2021, ISBN 978-1-6654-1156-1, pp. 1056–1065. DOI ↑70, 71, 73

[110] *Advanced Vector Extensions 512 (AVX-512) — x86* (Accessed 12.05.2025). URL ↑70, 143

[111] M. Gottschlag, F. Bellosa. *Mechanism to mitigate AVX-induced frequency reduction*, 2018, 12 pp. arXiv 1901.04982 ↑71

[112] M. Gottschlag, P. Brantsch, F. Bellosa. "Automatic core specialization for AVX-512 applications", *SYSTOR '20: Proceedings of the 13th ACM International Systems and Storage Conference* (Haifa, Israel, 2–4 June 2020), ACM, New York, 2020, ISBN 978-1-4503-7588-7, pp. 25–35. doi ↑71

[113] M. Gottschlag, T. Schmidt, F. Bellosa. AVX overhead profiling: how much does your fast code slow you down? *APSys '20: Proceedings of the 11th ACM SIGOPS Asia-Pacific Workshop on Systems* (Tsukuba Japan, 24–25 August 2020), ACM, New York, 2020, ISBN 978-1-4503-8069-0, pp. 59–66. doi ↑71

[114] J. M. Cebrian, L. Natvig, M. Jahre. "Scalability analysis of AVX-512 extensions", *The Journal of supercomputing*, **76**:3 (2020), pp. 2082–2097. doi ↑71

[115] *Ice Lake AVX-512 Downclocking*, 2020 (Accessed 12.05.2025). URL ↑71

[116] A. Frumusanu. *Intel 3rd Gen Xeon Scalable (Ice Lake SP) Review: Generationally Big, Competitively Small*, 2021 (Accessed 12.05.2025(**)). URL ↑71

[117] *3rd and 4th Gen Xeon Scalable Turbo Frequencies*, 2023 (Accessed 12.05.2025). URL ↑71

[118] *Accelerate Artificial Intelligence (AI) Workloads with Intel Advanced Matrix Extensions (Intel AMX) ID 785250*, 2024 (Accessed 25.03.2025(*)). URL ↑73, 204

[119] T. Burd, S. Venkataraman, W. Li, T. Johnson, J. Lee, S. Velaga. "2.2 "Zen 4c": The AMD 5nm area-optimized x86-64 microprocessor core", *2024 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 18–22 February 2024), IEEE, 2024, ISBN 979-8-3503-0621-7, pp. 38–40. doi ↑73

[120] *AMD EPYC™ 8004 Series Architecture Overview*, Revision 1.0 (Sept. 2023), 2023 (Accessed 12.05.2025), 14 pp. URL ↑74

[121] K. Mayo, S. Rajasekaran, I. Pasichnyk, A. Kashyap. *High Performance Computing Tuning Guide for AMD EPYC™ 9004 Series Processors Publication 58002*, Revision 1.5 Jan 2024, 2024 (Accessed 15.11.2024), 60 pp. URL ↑74, 77, 79, 83, 84, 85, 86, 87, 88, 92, 104, 105, 106, 109, 110, 111, 112, 124, 125, 150, 192, 195, 224

[122] *AMD EPYC™ 9004 Series Processors Data Sheet*, 2023 (Accessed 09.01.2026), 2 pp. URL ↑74, 78

[123] *SLES 15 SP4 Optimizing Linux for AMD EPYC™ 9004 Series Processors with SUSE* (Accessed 12.05.2025(**)). URL ↑74

[124] H. Mujtaba. *AMD's Monstrous EPYC™ Genoa CPU For SP5 "LGA 6096" Socket Pictured: Up To 96 Zen 4 Cores, 400W TDP*, 2022 (Accessed 20.05.2024). URL ↑75

[125] *AMD EPYC™ 8004 Series Processors Data Sheet*, 2023 (Accessed 12.05.2025), 2 pp. URL ↑76, 78

[126] *AMD Server Processor Specifications* (Accessed 15.10.2024). URL ↑78, 232, 233

[127] T. P. Morgan. *Why AMD "Genoa" Epyc Server CPUs Take The Heavyweight Title*, 2022 (Accessed 26.05.2024). URL ↑81, 91

[128] *Infinity Fabric (IF) — AMD* (Accessed 3.09.2024). URL ↑81

[129] S. Lehmann, F. Gerfers. "Channel analysis for a 6.4 Gb/s DDR5 data buffer receiver front-end", *2017 15th IEEE International New Circuits and Systems Conference (NEWCAS)* (Strasbourg, France, 25–28 June 2017), IEEE, 2017, ISBN 9781728110325, pp. 109–112. doi ↑84

[130] *Micron's Perspective on Impact of CXL on DRAM Bit Growth Rate*, 2023 (Accessed 12.05.2025), 9 pp. URL ↑84, 86

[131] Y. Tang, P. Zhou, W. Zhang, H. Hu, Q. Yang, Q. Xiang, T. Liu, J. Shan, R. Huang, Ch. Zhao, Ch. Chen, H. Zhang, F. Liu, Sh. Zhang, X. Ding, J. Chen. "Exploring performance and cost optimization with ASIC-based CXL memory", *EuroSys '24: Proceedings of the Nineteenth European Conference on Computer Systems* (Athens, Greece, 22–25 April 2024), ACM, New York, 2024, ISBN 979-8-4007-0437-6, pp. 818–833. doi ↑84

[132] *5-Level Paging and 5-Level EPT*, White Paper Revision 1.1, 2017 (Accessed 09.01.2026), 31 pp. URL ↑88

[133] *AMD Infinity Guard*, 2024 (Accessed 1.06.2024). URL ↑89

[134] *Processor Family with Full Memory Encryption as a Standard Security Feature*, 2020 (Accessed 3.08.2024). URL ↑89

[135] *AMD Secure Encrypted Virtualization (SEV)*, 2024 (Accessed 1.06.2024). URL ↑89

[136] H. N. Jacob, Ch. Werling, R. Buhren, J.-P. Seifert. "faulTPM: Exposing AMD fTPMs' deepest secrets", *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (Delft, Netherlands, 03–07 July 2023), IEEE, 2023, ISBN 9781665465137, pp. 1128–1142. doi ↑89

[137] L. Wilke, J. Wichelmann, M. Morbitzer, Th. Eisenbarth. "SEVurity: No security without integrity: Breaking integrity-free memory encryption with minimal assumptions", *2020 IEEE Symposium on Security and Privacy (SP)* (San Francisco, California, USA, 18–21 May 2020), IEEE, 2020, ISBN 978-1-7281-3498-7, pp. 1483–1496. doi URL ↑89

[138] M. S. Karvandi, S. Meghdadizanjani, S. Arasteh, S. Kh. Monfared, M. K. Fallah, S. Gorgin, J.-A. Lee, E. van der Kouwe. *The reversing machine: reconstructing memory assumptions*, 2024, 17 pp. arXiv 2405.00298 ↑89

[139] *AMD EPYC™ 4004 Series Processors Data Sheet*, 2024 (Accessed 1.06.2024), 2 pp. URL ↑90

[140] *Artificial Intelligence Machine Learning (AI/ML) Tuning Guide for AMD EPYC™ 9004 Series Processors*, Rev.1.1, 2023 (Accessed 26.11.2024). URL ↑92, 132

[141] *Lenovo ThinkAgile VX655 V3 2U Certified Node (AMD EPYC™ 9004) Product Guide*, 2024 (Accessed 3.08.2024), 61 pp. URL ↑92

[142] *Configuring AMD xGMI Links on the Lenovo ThinkSystem SR665 V3 Server*, 2024 (Accessed 3.08.2024), 15 pp. URL ↑92, 93

[143] *Compute Nodes*, 2024 (Accessed 21.08.2024). URL ↑93

[144] S. Atchley, Ch. Zimmer, J. Lange, D. Bernholdt, V. M. Vergara, Th. Beck, M. Brim, R. Budiardja, S. Chandrasekaran, M. Eisenbach, Th., Evans Ezell M., Frontiere N., Georgiadou A., Glenski J., Grete Ph., Hamilton S., Holmen J., Huebl A., Jacobson D., Joubert W., Mcmahon K., Merzari E., Moore S., Myers A., Nichols S., Oral S., Papatheodore Th., Perez D., Rogers D. M., Schneider E., Vay J.-L., Yeung P. K.. "Frontier: exploring exascale", *SC '23: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ACM, New York, 2023, ISBN 979-8-4007-0109-2, id. 52, 16 pp. doi ↑93, 243, 244

[145] *HoreKa Hardware Overview*, 2024 (Accessed 2.08.2024). URL ↑93, 244

[146] *AMD Documentation Hub* (Accessed 11.11.2024). URL ↑94, 234

[147] M. Larabel. *AMD EPYC™ 9554 & EPYC™ 9654 Benchmarks — Outstanding Performance For Linux HPC/Servers*, 2022 (Accessed 20.10.2024), 15 pp. URL ↑95, 109, 122, 123, 241

[148] R. Nambiar. *AMD Ecosystem Poised and Ready for the 4th Gen AMD EPYC™ Processors*, 2022 (Accessed 13.10.2025). URL ↑95, 109, 110, 139

[149] M. Larabel. *AlmaLinux, CentOS Stream, Clear Linux, Debian, Fedora & Ubuntu On AMD 4th Gen EPYC™ Genoa*, 2023 (Accessed 20.10.2024), 6 pp. URL ↑95

[150] M. Larabel. *AMD 4th Gen EPYC™ 9654 "Genoa" AVX-512 Performance Analysis*, 2022 (Accessed 11.11.2024), 9 pp. URL ↑96, 131, 132

[151] W. C. Lin, S. McIntosh-Smith. "Comparing Julia to performance portable parallel programming models for HPC", *2021 International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS)* (St. Louis, MO, USA, 15 November 2021), IEEE, 2021, ISBN 978-1-6654-1119-6, pp. 94–105. doi ↑96

[152] *AMD EPYC™ 9004 Series Performance Package*, 2022 (Accessed 9.10.2024), 35 pp. URL ↑96, 110, 111

[153] *SPEC OMP 2012 Results* (Accessed 13.11.2024). URL ↑97

[154] *Tuning SPEComp2012 Performance on Lenovo ThinkSystem AMD Servers*, 2023 (Accessed 20.10.2024), 14 pp. URL ↑99

[155] *AMD Epyc 9004 "Genoa" buyers guide for CFD*, 2022 (Accessed 21.10.2024). URL ↑98, 99, 118, 119

[156] *All Published SPEC MPIM 2007 Results* (Accessed 6.05.2025). URL ↑99

[157] M. Larabel. *Ampere Altra Max 128-Core CPU Is Priced Lower Than Flagship Xeon, EPYC™ CPUs*, 2021 (Accessed 12.05.2025). URL ↑102

[158] K. Yalamanchi, S. Vazhkudai. *Boost HPC workloads with Micron DDR5 and 4th Gen AMD EPYC™ processors*, 2022 (Accessed 9.02.2025). ↑103

[159] A. Rajasukumar, T. Zhang, R. Xu, A. A. Chien. "UpDown: a novel architecture for unlimited memory parallelism", *MEMSYS '24: The International Symposium on Memory Systems* (Washington, DC, USA, 30 September 2024–3 October 2024), ACM, New York, 2024, ISBN 979-8-4007-1091-9, pp. 61–77. ↑106

[160] R. Nambiar. *Performance Analysis of HPC Workloads: AMD EPYC™ with 3D V-Cache vs. Intel Xeon CPU Max with HBM*, 2023 (Accessed 6.12.2024). ↑129, 130, 221

[161] *Fujitsu Server PRIMERGY Memory performance of EPYC™ 9004 Series Processor (Genoa) based Systems White Paper v. 1.0*, 2024 (Accessed 18.09.2024). ↑107

[162] *Workload-Based DDR5 Memory Guidance for Next-Generation PowerEdge Servers*, 2023 (Accessed 19.12.2024), 21 pp. ↑107, 108, 139

[163] I. Z. Reguly. "Comparative evaluation of bandwidth-bound applications on the Intel Xeon CPU MAX Series", *SC-W '23: Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0785-8, pp. 1236–1244. ↑108, 217, 221

[164] *Fujitsu Server PRIMERGY Performance Report. PRIMERGY RX2530 M6*, 2023 (Accessed 11.02.2025). ↑110

[165] M. Sztemon. *AMD Data Center Portfolio*, 2024 (Accessed 9.10.2024), 55 pp. ↑111, 121, 139

[166] *Fujitsu Server Performance Report PRIMERGY RX2530 M7 / RX2540 M7 v.1.5*, 2024 (Accessed 31.12.2024). ↑111, 144, 178, 187, 192, 198

[167] J. Löff, D. Griebler, G. Mencagli, G. Araujo, M. Torquati, M. Danelutto, L. G. Fernandes. "The NAS parallel benchmarks for evaluating C++ parallel programming frameworks on shared-memory architectures", *Future Generation Computer Systems*, **125**:C (2021), pp. 743–757. ↑112

[168] V. Stegailov, E. Dlinnova, T. Ismagilov, M. Khalilov, N. Kondratyuk, D. Makagon, A. Semenov, A. Simonov, G. Smirnov, A. Timofeev. "Angara interconnect makes GPU-based Desmos supercomputer an efficient tool for molecular dynamics calculations", *The International Journal of High Performance Computing Applications*, **33**:3 (2019), pp. 507–521. ↑113

[169] A. Ayala, S. Tomov, A. Haidar, J. Dongarra. "*heFFTe*: Highly efficient FFT for exascale", *Computational Science — ICCS 2020: 20th International Conference*. V. I (Amsterdam, The Netherlands, 3–5 June 2020), Springer-Verlag, Berlin–Heidelberg, 2020, ISBN 978-3-030-50370-3, pp. 262–275. ↑115

[170] A. Ayala, S. Tomov, M. Stoyanov, J. Dongarra. "Scalability issues in FFT computation", *Parallel Computing Technologies: 16th International Conference*, PaCT 2021 (Kaliningrad, Russia, 13–18 September 2021), Lecture Notes in Computer Science, vol. **12942**, Springer, Cham, 2021, ISBN 978-3-030-86358-6, pp. 279–287. ↑116

[171] M. Larabel. *AMD EPYC™ 9684X Genoa-X Provides Incredible HPC Performance*, 2023 (Accessed 25.11.2024), 8 pp. URL ↑116, 119, 122, 123, 134

[172] G. Moon, M. Yi, E. Jun. "Design and operational concept of a cryogenic active intake device for atmosphere-breathing electric propulsion", *Aerospace Science and Technology*, **151** (2024), id. 109300. doi ↑117

[173] P. German, O. Marin, G. L. Giudicelli, J. E. Hansel, A. D. Lindsay, D. C., Yankura Charlot L., Freile R. O., Tano M. E.. *Continued performance improvement and integration of MOOSE's thermal-hydraulics capabilities*, M3 Milestone Report, NoINL/RPT-24-80844-Rev000, Idaho National Laboratory (INL), Idaho Falls, ID, USA, 2024 (Accessed 29.11.2024), 44 pp. URL ↑117

[174] B. A. Danciu, C. E. Frouzakis. *KinetiX: A performance portable code generator for chemical kinetics and transport properties*, 2024, 34 pp. arXiv 2411.02640 ↑117

[175] S. Gross. *Simulation Hardware: The Sky is the Limit?* 2022 (Accessed 26.11.2024). URL ↑118

[176] *OpenFOAM v12 User Guide*, 2024 (Accessed 29.11.2024). URL ↑118

[177] I. Spisso, S. Bna, G. Amati, G. Rossi, F. Magugliani. *HPC Benchmark Project (part I)*, OpenFoam Conference (Germany, 15–17 October 2019), 2019 (Accessed 29.11.2024), 27 pp. URL ↑119

[178] *OpenFOAM HPC Benchmark Suite*, 2019 (Accessed 29.11.2024). URL ↑119

[179] *OpenFOAM and AMD 3d V-cache Technology Computational Fluid Dynamics*, 2023 (Accessed 8.12.2024). URL ↑119

[180] F. C. C. Galeazzo, F. Zhang, Th. Zirwes, P. Habisreuther, H. Bockhorn, N. Zarzalis, D. Trimis. "Implementation of an efficient synthetic inflow turbulence-generator in the open-source code OpenFOAM for 3D LES/DNS applications", *High Performance Computing in Science and Engineering'20: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2020*, Springer, Cham, 2021, ISBN 978-3-030-80601-9, pp. 207–221. doi ↑119

[181] F. C. C. Galeazzo, R. G. Weiß, S. Lesnik, H. Rusche, A. Ruopp. *Understanding Superlinear Speedup in Current HPC Architectures*, 2024 (Accessed 30.11.2024), 9 pp. doi URL ↑119

[182] X. Álvarez-Farré, A. Gorobets, F. X. Trias. "A hierarchical parallel implementation for heterogeneous computing. Application to algebra-based CFD simulations on hybrid supercomputers", *Computers & Fluids*, **214** (2021), id. 104768. doi ↑119

[183] S. Pareek, K. Veena, M. Naik, P. Donthireddy. *HPC Application Performance on Dell PowerEdge R6625 with AMD EPYC™-GENOA*, 2023 (Accessed 30.11.2024). URL ↑119, 123, 124, 126, 128, 131

[184] *Supermicro, AMD, WEKA and NVIDIA deliver High Performance Ansys Turnkay Solution*, 2024 (Accessed 5.10.2024), 11 pp. URL ↑120

[185] F. Banchelli, M. Garcia-Gasulla, G. Houzeaux, F. Mantovani. "Benchmarking of state-of-the-art HPC Clusters with a Production CFD Code", *PASC '20: Proceedings of the Platform for Advanced Scientific Computing Conference* (Geneva, Switzerland, 29 June 2020–1 July 2020), ACM, New York, 2020, ISBN 978-1-4503-7993-9, id. 3, 11 pp. ↑120

[186] A. Fernandez. *Ansys Applications Technical Computing*, 2022 (Accessed 8.05.2025). ↑120

[187] A. et al. Fernandez. *Leadership Performance on ANSYS Applicattions Finite Element Analysis & Computational Fluid Dynamics*, 2024 (Accessed 8.12.2024(**)). ↑120, 121

[188] A. Fernandez, A. Manikonda. *ANSYS LS-DYNA and AMD 3d V-cache Technology*, 2023 (Accessed 11.10.2024). ↑121

[189] A. Fernandez, A. Manikonda. *ANSYS CFX and AMD 3D V-Cache Technology Computational Fluid Dynamics*, 2023 (Accessed 8.12.2024). ↑121

[190] A. Fernandez, A. Manikonda. *ANSYS FLUENT and AMD 3D V-Cache Technology Computational Fluid Dynamics*, 2023 (Accessed 11.10.2024). ↑121

[191] B. Wilfong, A. Radhakrishnan, H. A. Le Berre, S. Abbott, R. D. Budiardja, S. H. Bryngelson. *OpenACC offloading of the MFC compressible multiphase flow solver on AMD and NVIDIA GPUs*, 2024, 11 pp. arXiv 2409.10729 ↑122

[192] A. Fernandez, A. Manikonda. *WRF and AMD 3D V-cache technology weather forecasting*, 2023 (Accessed 8.05.2025). ↑122, 123

[193] M. Larabel. *AMD EPYC™ 9374F Linux Benchmarks — Genoa's 32-Core High Frequency CPU*, 2022 (Accessed 28.09.2024), 14 pp. ↑123

[194] S. Kashyap, S. Kovouri, P. Goyal. *GROMACS on Amazon EC2 Hpc7a Instances*, 2023 (Accessed 9.12.2024). ↑124

[195] A. Fernandez, A. Manikonda. *Molecular Dynamics on AMD EPYC™ 9754 Processors*, 2023 (Accessed 9.12.2024). ↑124, 129, 202, 203

[196] H. Malik, T. Nunes. *4th Gen AMD EPYC™ Processors for Healthcare and Life Science*, 2024 (Accessed 22.10.2024), 13 pp. ↑129, 132, 139, 202

[197] K. Ramakrishna, M. Lokamani, A. Cangi. *Electrical conductivity of warm dense hydrogen from Ohm's law and time-dependent density functional theory*, 2024, 9 pp. arXiv 2409.15160 ↑130

[198] Voorhis T., Vázquez-Mayagoitia Á., Verma P., Villa O., Vishnu A., Vogiatzis K. D., Wang D., Weare J. H., Williamson M. J., Windus T. L., Woliński K., Wong A. T., Wu Q., Yang C., Yu Q., Zacharias M., Zhang Z., Zhao Y., Harrison R. J.. "NWChem: Past, present, and future", *The Journal of chemical physics*, **152**:18 (2020), id. 184102. ↑131

[199] M. Larabel. *AMD EPYC™ 9374F Linux Benchmarks — Genoa's 32-Core High Frequency CPU*, 2022 (Accessed 18.05.2025), 14 pp. ↑131

[200] M. Larabel. *AMD EPYC™ 9755 / 9575F / 9965 Benchmarks Show Dominating Performance*, 2024 (Accessed 14.12.2024), 14 pp. ↑131, 209, 233, 236, 237, 238, 239

[201] M. Larabel. *AVX-512 Performance Comparison: AMD Genoa vs. Intel Sapphire Rapids & Ice Lake*, 2023 (Accessed 15.12.2024), 8 pp. URL ↑131

[202] M. Larabel. *Intel Xeon Platinum 8592+ "Emerald Rapids" Linux Benchmarks*, 2023 (Accessed 15.12.2024), 10 pp. URL ↑132, 194, 195, 196, 197, 198, 199, 201, 202, 203, 204, 208, 211

[203] J. Hof, D. Speed. "LDAK-KVIK performs fast and powerful mixed-model association analysis of quantitative and binary phenotypes", *Nature Genetics*, **57** (2025), pp. 2116–2123. doi ↑132

[204] G. Bernardini, L. van Iersel, E. Julien, L. Stougie. "Inferring phylogenetic networks from multifurcating trees via cherry picking and machine learning", *Molecular Phylogenetics and Evolution*, **199** (2024), id. 108137. doi ↑132

[205] C. M. Williams, J. O'Connell, W. A. Freyman, Research Team 23andMe, Ch. R. Gignoux, S. Ramachandran, A. L. Williams. *Phasing millions of samples achieves near perfect accuracy, enabling parent-of-origin classification of variants*, bioRxiv, 2024. doi ↑132

[206] A. McKenna, M. Hanna, E. Banks, A. Sivachenko, K. Cibulskis, A. Kernytsky, K. Garimella, D. Altshuler, S. Gabriel, M. Daly, M. A. DePristo. "The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data", *Genome research*, **20**:9 (2010), pp. 1297–1303. doi ↑132

[207] K. I. Kendig, S. Baheti, M. A. Bockol, T. M., Drucker Hart S. N., Heldenbrand J. R., Hernaez M., Hudson M. E., Kalmbach M. T., Klee E. W., Mattson N. R., Ross Ch. A., Taschuk M., Wieben E. D., Wiepert M., Wildman D. E., Mainzer L. S.. "Sentieon DNASeq variant calling workflow demonstrates strong computational performance and accuracy", *Frontiers in genetics*, **10** (2019), id. 736, 7 pp. doi ↑132

[208] H. Malik, M. Seniziaz, D. Freed, B. Gallagher. *4th Gen AMD EPYC™ Priocessor Accelerate Genomics Data Processing with Sentieon*, 2023 (Accessed 5.10.2024). URL ↑132

[209] *Nvidia Clara Parabrics Pipelines*, 2021 (Accessed 15.12.2024). URL ↑132

[210] *AI Inferencing with AMD EPYC™ Processors*, White Paper, 2024 (Accessed 09.01.2026), 11 pp. URL ↑132, 133

[211] D. M. F. Ha, T. Alderliesten, P. A. N. Bosman. "Learning discretized Bayesian networks with GOMEA", *Parallel Problem Solving from Nature — PPSN XVIII*. V. III, PPSN 2024 (Hagenberg, Austria, 14–18 September 2024), Lecture Notes in Computer Science, vol. **15150**, Springer, Cham, 2024, ISBN 978-3-031-70070-5, pp. 352–368. doi ↑132

[212] K. Nair, A.-Ch. Pandey, S. Karabannavar, M. Arunachalam, J. Kalamatianos, V. Agrawal, S. Gupta, A. Sirasao, E., Delaye Reinhardt S., Vivekanandham R., Wittig R., Kathail V., Gopalakrishnan P., Pareek S., Jain R., Kandemir M. T., Lin J.-L., Akbulut G. G., Das Ch. R.. "Parallelization Strategies for DLRM Embedding Bag Operator on AMD CPUs", *IEEE Micro*, **44**:6 (2024), pp. 44–51. doi ↑132

[213] G., Jocher Chaurasia A., Stoken A., Borovec J., NanoCode012, Kwon Y., Michael K., TaoXie, Fang J., Imyhxy, Lorna, Zeng Y., Wong C., Abhiram V., Montes D., Wang Zh., Fati C., Nadar J., Laughing, UnglvKitDe, Sonck V., Tkianai, YxNONG, Skalski P., Hogan A., Nair D., Strobel M., Jain M.. *ultralytics/yolov5: v7.0 — YOLOv5 SOTA. Realtime instance segmentation*, Zenodo, 2022. ↑133

[214] R. Nambiar. *4th Gen AMD EPYC™ Processors Deliver Exceptional Performance for AI Workloads*, 2023 (Accessed 27.11.2024). ↑133

[215] V. J. Reddi, Ch. Cheng, D. Kanter, P. Mattson, G. Schmuelling, C.-J. Wu, B. Anderson, M. Breughe, M. Charlebois, W. Chou, R. Chukka, C. Coleman, S. Davis, P. Deng, G., Diamos Duke J., Fick D., Gardner J. S., Hubara I., Idgunji S., Jablin Th. B., Jiao J., John T. St., Kanwar P., Lee D., Liao J., Lokhmotov A., Massa F., Meng P., Micikevicius P., Osborne C., Pekhimenko G., Rajan A. T. R., Sequeira D., Sirasao A., Sun F., Tang H., Thomson M., Wei F., Wu E., Xu L., Yamada K., Yu B., Yuan G., Zhong A., Zhang P., Zhou Yu.. "MLPerf inference benchmark", *ISCA '20: Proceedings of the ACM/IEEE 47th Annual International Symposium on Computer Architecture* (Virtual Event, 30 May 2020–3 June 2020), IEEE, 2020, ISBN 978-1-7281-4661-4, pp. 446–459. ↑133, 205, 206

[216] Y. Gorbachev, Yu. Gorbachev, M. Fedorov, I. Slavutin, A. Tugarev, M. Fatekhov. "OpenVINO deep learning workbench: Comprehensive analysis and tuning of neural networks inference", *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (Seoul, Korea (South), 27–28 October 2019), IEEE, 2019, ISBN 9798350307450, pp. 783–787 (Accessed 27.05.2025). ↑133

[217] M. Larabel. *Intel Xeon Platinum 8490H "Sapphire Rapids" Performance Benchmarks*, 2023 (Accessed 23.04.2025), 14 pp. ↑134

[218] J. Rasley, S. Rajbhandari, O. Ruwase, Yu. He. "DeepSpeed: System optimizations enable training deep learning models with over 100 billion parameters", *KDD '20: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Virtual Event, CA, USA, 6–10 July 2020), ACM, New York, 2020, ISBN 978-1-4503-7998-4, pp. 3505–3506. ↑134

[219] S. H. Kim, X. Wang, Y. Fu. *Practical Strategies for low-cost LLM Deployments using 4th Gen AMD EPYC™ Processors*, 2024 (Accessed 28.11.2024), 9 pp. ↑134, 135

[220] *TPCx-AI Specification Version 2.0.0*, 2024 (Accessed 27.11.2024), 85 pp. ↑136

[221] A. Rajput. *AMD EPYC™ 9004 Tunuing guide. Cloud Infrastructure and Datacenter Design & Configuration Rev. 1.03*, 2024 (Accessed 18.12.2024), 64 pp. ↑137, 230

[222] G. Gibby. *Meet the New AMD EPYC™ 97X4 Processors, Optimized for Cloud-Native Workloads*, 2023 (Accessed 18.12.2024). ↑137

[223] R. Nambiar. *"Bergamo" 4th Gen AMD EPYC™ 97x4 Processors: Built for Cloud Native Workloads*, 2023 (Accessed 18.12.2024). URL ↑137, 139
[224] *VMware Marketplace*, 2024 (Accessed 15.12.2024). URL ↑137
[225] *VMmark 3.x Results*, 2024 (Accessed 15.12.2024). URL ↑137
[226] *VMmark 4 Results*, 2024 (Accessed 15.12.2024). URL ↑137
[227] *PostgreSQL pgbench*, 2024 (Accessed 16.12.2024). URL ↑138
[228] *AMD EPYC™ Delivers Leadership MariaDB Perforamnce Relational Database*, 2024 (Accessed 17.10.2024). URL ↑139
[229] G. Rajaram, J. Porana, B. Veerabhadrachary, D. Kenchappanavara. *Cloud-Native Workloads on AMD EPYC™ 9754 Processors*, 2023 (Accessed 09.01.2026), 13 pp. URL ↑139
[230] R. Nambiar. *4th Gen AMD EPYC™ 8004 Series Processors Designed for Intelligent Edge*, 2023 (Accessed 18.12.2024). URL ↑140
[231] *Intel Xeon Platinum 8380 Processor* (Accessed 5.08.2024(*)). URL ↑141
[232] H. N. Schuh, A. Krishnamurthy, D. Culler, H. M. Levy, L. Rizzo, S. Khan, B. E. Stephens. "CC-NIC: a cache-coherent interface to the NIC", *ASPLOS '24: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*. V. 1, ACM, New York, 2024, ISBN 979-8-4007-0372-0, pp. 52–68. doi ↑142
[233] D. L. Mulnix. *Intel Xeon Processor Scalable Family Technical Overview*, 2022 (Accessed 5.08.2024(*)). URL ↑143, 145, 146, 147
[234] *Product Naming Convention for the 4th and 5th Gen Intel Xeon Scalable Processors*, 2024 (Accessed 10.08.2024(*)). URL ↑144
[235] *Intel Xeon Scalable Processors Numbers and Suffixes*, 2024 (Accessed 13.01.2025(*)). URL ↑143, 144
[236] D. Watts. *Lenovo ThinkSystem SR950 V3 Server Product Guide*, 2025 (Accessed 09.01.2026), 80 pp. URL ↑145
[237] *Intel Scalable I/O Virtualization*, 2020 (Accessed 09.01.2026), 29 pp. URL ↑147
[238] N. Nassif, A. O. Munch, C. L. Molnar, G. Pasdast, S. V. Lyer, Z. Yang. "Sapphire Rapids: The next-generation Intel Xeon Scalable processor", *2022 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 20–26 February 2022), IEEE, 2022, ISBN 978-1-6654-2801-9, pp. 44–46. doi ↑147, 151, 155, 156
[239] A. Biswas. *Sapphire Rapids*, HotChips, 2021 (Accessed 12.05.2025), 22 pp. URLIntel Arijit.pdf ↑147, 148, 156
[240] *Network-Optimized 4th Gen Intel Xeon Scalable Processors Product Brief* (Accessed 5.08.2024(*)). URL ↑147
[241] *Intel® 64 and IA-32 Architectures Optimization Reference Manual Documentation Changes*, Aug 2023, 2023 (Accessed 5.08.2024), 379 pp. URL ↑148, 149, 150, 151
[242] H. Mujtaba. *AMD Zen 4 For Ryzen 7000 & Intel Raptor Cove For Raptor Lake CPUs Have Almost Similar IPC*, 2022 (Accessed 15.02.2025). URL ↑150

[243] D. et al. Mulnix. *Technical Overview Of The 4th Gen Intel Xeon Scalable processor family*, 2022 (Accessed 5.08.2024(*)). ↑151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 162

[244] *Intel 64 and IA-32 Architectures Software Developer Manuals Upd.* (Accessed 5.08.2024(*)). ↑151

[245] *Intel Architecture Instruction Set Extensions and Future Features Programming Reference*, 2024 (Accessed 6.08.2024(*)). ↑152, 153, 154

[246] *Intel 64 and IA-32 Architectures Optimization Reference Manual.* V. 1, 2023 (Accessed 09.01.2026), 963 pp. ↑153, 154

[247] F. Dizani, A. Ghanbari, J. Kalyanapu, D. Asher, S. M. Ajorpaz. "Thor: A non-speculative value dependent timing side channel attack exploiting Intel AMX", *IEEE Computer Architecture Letters*, **24**:1 (2025), pp. 69–72. ↑153

[248] I. Cutress. *Intel Xeon Sapphire Rapids: How To Go Monolithic with Tiles*, 2021 (Accessed 23.04.2025). ↑155

[249] R. Mahajan, R. Sankman, N. Patel, D.-W. Kim, K. Aygun, Zh. Qian. "Embedded multi-die interconnect bridge (EMIB) — a high density, high bandwidth packaging interconnect", *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)* (Las Vegas, NV, USA, 31 May 2016–03 June 2016), IEEE, 2016, ISBN 9781509012053, pp. 557–565. ↑156

[250] R. Mahajan, R. Sankman, K. Aygun, Zh. Qian, A. Dhall, J. Rosch, D. Mallik, I. Salama. "Embedded Multi-die Interconnect Bridge (EMIB). A localized, high density, high bandwidth packaging interconnect", *Advances in Embedded and Fan-Out Wafer-Level Packaging Technologies*, Ch. 23, eds. Keser B., Kroehnert S., Wiley, 2019, ISBN 9781119314134, pp. 487–499. ↑156

[251] G. Duan, Y. Kanaoka, R. McRee, B. Nie, R. Manepalli. "Die embedding challenges for EMIB advanced packaging technology", *2021 IEEE 71st Electronic Components and Technology Conference (ECTC)* (San Diego, CA, USA, 01 June 2021–04 July 2021), IEEE, 2021, ISBN 9781665431200, pp. 1–7. ↑156

[252] C. C. Lee, Y. W. Chang. "Floorplanning for embedded multi-die interconnect bridge packages", *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)* (San Francisco, CA, USA, 28 October 2023–02 November 2023), IEEE, 2023, ISBN 9798350322262, pp. 1–8. ↑156

[253] H. Mujtaba. *Intel Sapphire Rapids-SP Xeon Server CPU Detailed: Quad-Tile Chiplet Design With EMIB, 56 Cores, 112 Threads, CXL 1.1, DDR5, HBM & PCIe 5.0 Support*, 2021 (Accessed 10.05.2025). ↑156

[254] R. Kuper, I. Jeong, Y. Yuan, R. Wang, N. Ranganathan, N. Rao, J. Hu, S. Kumar, P. Lantz, N. S. Kim. "A quantitative analysis and guidelines of data streaming accelerator in modern Intel Xeon Scalable Processors", *ASPLOS '24: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems.* V. 2 (La Jolla, CA, USA, 27 April 2024–1 May 2024), ACM, New York, 2024, ISBN 979-8-4007-0385-0, pp. 37–54. ↑159

[255] A. Berthold, C. Fürst, A. Obersteiner, L. Schmidt, D. Habich, W. Lehner, H. Schirmeier. “Demystifying Intel Data Streaming Accelerator for in-memory data processing”, *DIMES ’24: Proceedings of the 2nd Workshop on Disruptive Memory Systems* (Austin, TX, USA, 3 November 2024), ACM, New York, 2024, ISBN 979-8-4007-1303-3, pp. 9–16. ↑159

[256] *Proof Points of Intel® Dynamic Load Balancer (Intel® DLB)* (Accessed 10.08.2024(*)). ↑159

[257] R. Sexton, D. Coyle, D. Przychodni. *Guidelines for Optimizing Power Consumption of vCMTS Deployments on Intel Xeon Scalable Processors* (Accessed 19.08.2024), 15 pp. ↑160, 161

[258] R. Pattan, H. Khan. *Intel Turbo Boost Technology Configure Per Core Turbo Overview*, 2022 (Accessed 24.04.2025), 6 pp. ↑161

[259] J. Huffstetler. *Sustainability In Focus with 4th Gen Intel Xeon Processors*, 2023 (Accessed 19.03.2025). ↑161

[260] E. Rieger. *Half-Socket VM support for SAP HANA on vSphere 8 and 4th Gen Intel® Xeon® Scalable Processors (Sapphire Rapids)*, 2024 (Accessed 13.08.2024). ↑161

[261] *Intel® Virtualization Technology for Directed I/O Architecture Specification*, Rev. 4.1, 2023 (Accessed 13.08.2024(*)). ↑162

[262] D. Moghimi. “Downfall: Exploiting speculative data gathering”, *SEC ’23: Proceedings of the 32nd USENIX Conference on Security Symposium* (Anaheim, CA, USA, 9–11 August 2023), USENIX Ass., Berkeley, 2023, ISBN 978-1-939133-37-3, pp. 7179–7193, id. 402. ↑163

[263] *Intel® Trust Domain Extensions*, White Paper, 2022 (Accessed 14.08.2024), 9 pp. ↑163, 164, 165, 166

[264] A. Sunny, N. Shrivastava, S. R. Sarangi. *SecScale: a scalable and secure trusted execution environment for servers*, 2024, 13 pp. arXiv 2407.13572 ↑163, 164, 166

[265] A. Akram, A. Giannakou, V. Akella, J. Lowe-Power, S. Peisert. “Performance analysis of scientific computing workloads on general purpose TEEs”, *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (Portland, OR, USA, 17–21 May 2021), IEEE, 2021, ISBN 978-1-6654-1156-1, pp. 1066–1076. ↑163, 164

[266] D. Miladinović, A. Milaković, M., Vukasović, Ž. Stanisavljević, P. Vuletić. “Secure Multiparty Computation Using Secure Virtual Machines”, *Electronics*, **13**:5 (2024), id. 991, 25 pp. ↑163, 164

[267] O. Demigha, R. Larguet. “Hardware-based solutions for trusted cloud computing”, *Computers & Security*, **103** (2021), id. 102117. ↑164, 166

[268] M. Li, L. Wilke, J. Wichelmann, Th. Eisenbarth, R. Teodorescu, Y. Zhang. “A systematic look at ciphertext side channels on AMD SEV-SNP”, *2022 IEEE Symposium on Security and Privacy (SP)* (San Francisco, CA, USA, 22–26 May 2022), IEEE, 2022, ISBN 9781665413176, pp. 337–351. ↑164

[269] *Intel® Trust Domain Extension Research and Assurance*, Version 2.0, 2024 (Accessed 15.08.2024(*)). URL ↑164, 165

[270] W. Wang, L. Song, B. Mei, Sh. Liu, Sh. Zhao, Sh. Yan, X.-F. Wang, D. Meng, R. Hou. *NestedSGX: bootstrapping trust to enclaves within confidential VMs*, 2024, 16 pp. arXiv 2402.11438 ↑166

[271] C. Stephan, R. Rahman. *Balanced Memory Configurations for 2-Socket Servers with 4th and 5th Gen Intel Xeon Scalable Processors*, 2024 (Accessed 21.08.2024), 13 pp. URL ↑166

[272] S. et al. Lesmanne. *Rise of Volumetric Data and Scale-Up Enterprise Computing*, 2021 (Accessed 8.08.2024(**)). URL ↑166

[273] *HPE Compute Scale-up Server 3200*, 2023 (Accessed 10.08.2024), 7 pp. URL ↑167

[274] *HPE Compute Scale-up 3200 Server Performance Tuning Guide* (Accessed 12.05.2025). URL ↑167

[275] *Microsoft Eagle ND v5 System* (Accessed 21.08.2024). URL ↑167

[276] M. Turisini, G. Amati, M. Cestari. *LEONARDO: A pan-European pre-exascale supercomputer for HPC and AI applications*, 2023, 16 pp. arXiv 2307.16885 ↑167

[277] *MareNostrum 5 System overview* (Accessed 21.08.2024). URL ↑167

[278] G. M. Shipman, S. Swaminarayan, G. Grider, J. Lujan, R. J. Zerr. *Early Performance Results on 4th Gen Intel® Xeon® Scalable Processors with DDR and Intel® Xeon® processors, codenamed Sapphire Rapids with HBM*, 2022, 5 pp. arXiv 2211.05712 ↑168, 173, 217

[279] *Technical Overview Of The Intel Xeon Scalable processor Max Series*, 2022 (Accessed 11.08.2024(*)). URL ↑169, 170

[280] S. Kuo, J. Cheng. *Implementing High Bandwidth Memory and Intel Xeon Processors Max Series on Lenovo ThinkSystem Servers*, 2024 (Accessed 24.09.2024), 21 pp. URL ↑168, 170, 171, 218, 219

[281] J. D. McCalpin. "Bandwidth limits in the Intel Xeon Max (Sapphire Rapids with HBM) processors", *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, 21–25 May 2023), Lecture Notes in Computer Science, vol. **13999**, Springer, Cham, 2023, ISBN 978-3-031-40842-7, pp. 403–413. doi ↑169, 219, 220, 221, 229

[282] *Intel® Xeon® CPU Max Series Configuration and Tuning Guide*, 2023 (Accessed 12.08.2024), 35 pp. URL ↑170

[283] K. Fukazawa, R. Takahashi. "Performance evaluation of the fourth-generation Xeon with different memory characteristics", *HPCAsia '24 Workshops: Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region Workshops* (Nagoya, Japan, 25–27 January 2024), ACM, New York, 2024, ISBN 979-8-4007-1652-2, pp. 55–62. doi ↑171, 189, 217, 219, 220, 225, 226

[284] V. Sanca, A. Ailamaki. "Post-Moore's Law Fusion: high-bandwidth memory, accelerators, and native half-precision processing for CPU-local analytics", Joint Workshop sat 49th International Conference on Very Large Data Bases (VLDBW'23) — Workshop on Accelerating Analytics and Data Management Systems (ADMS'23) (Vancouver, Canada, August 28–September 1 2023), CEUR Workshop Proceedings, vol. **3462**, 2023, id. ADMS1 (Accessed 13.08.2024), 13 pp. URL ↑172, 217, 218

[285] A. O. Munch, N. Nassif, C. L. Molnar, J. Crop, R. Gammack, Ch. P. Joshi. "2.3 Emerald Rapids: 5th-Generation Intel® Xeon® Scalable Processors", *2024 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 18–22 February 2024), IEEE, 2024, ISBN 979-8-3503-0621-7, pp. 40–42. doi ↑173, 175, 176

[286] *5th Gen Intel ® Xeon ® Processors Product Brief*, 2023 (Accessed 18.08.2024(*)). URL ↑173, 175, 176, 177

[287] *5th Gen Intel Xeon Scalable Processor XCC (Codename Emerald Rapids) Uncore Performance Monitoring Guide*, Rev. 001, 2024 (Accessed 29.10.2024), 353 pp. URL ↑173

[288] C. Bai, J. Huang, X. Wei, Yu. Ma, S. Li, H. Zheng, B. Yu, Yu. Xie. "ArchExplorer: Microarchitecture exploration via bottleneck analysis", *MICRO '23: Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture* (Toronto, ON, Canada, 28 October 2023–1 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0329-4, pp. 268–282. doi ↑174

[289] P. Alcorn. *Intel 'Emerald Rapids' 5th-Gen Xeon Platinum 8592+ Review: 64 Cores, Tripled L3 Cache and Faster Memory Deliver Impressive AI Performance*, 2023 (Accessed 7.01.2025). URL ↑174, 175, 176, 177, 194, 200, 209

[290] *5th Gen Intel® Xeon® Scalable Processors* (Accessed 22.08.2024(*)). URL ↑175

[291] T. P. Morgan. *Intel "Emerald Rapids" Xeon SPs: A Little More Bang, A Little Less Bucks*, 2023 (Accessed 20.08.2024). URL ↑175, 177

[292] H. Mujtaba. *Intel 5th Gen Xeon CPUs Official: Emerald Rapids Compatible With Sapphire Rapids, Up To 64 Cores, 320 MB Cache, Prices Detailed*, 2023 (Accessed 17.08.2024). URL ↑176

[293] *Lenovo Launches New and Updated Servers with 5th Gen Intel Xeon Scalable Processors*, 2023 (Accessed 21.02.2025), 10 pp. URL ↑176

[294] *Performance Tuning Guide*, 2024 (Accessed 2.01.2024). URL ↑178

[295] *CPU Practice Guide*, 2024 (Accessed 2.01.2024). URL ↑178

[296] *4th Generation Intel Xeon Scalable Processors* (Accessed 4.02.2025(*)). URL ↑178

[297] *5th Generation Intel® Xeon® Scalable Processors* (Accessed 4.02.2025(*)). URL ↑178

[298] *Accelerate with Xeon*, 2023 (Accessed 22.12.2024), 69 pp. URL ↑178, 179

[299] A. Afzal, G. Hager, G. Wellein. "SPEChpc 2021 benchmarks on Ice Lake and Sapphire Rapids Infiniband clusters: A performance and energy case study", *SC-W '23: Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0785-8, pp. 1245–1254. ↑183, 184

[300] J. H. Laros III, K. Pedretti, S. M. Kelly, W. Shu, K. Ferreira, J. Vandyke, C. Vaughan. "Energy delay product", *Energy-Efficient High Performance Computing: Measurement and Tuning*, SpringerBriefs in Computer Science, Springer, London, 2013, ISBN 978-1-4471-4491-5, pp. 51–55. ↑183

[301] J. Laukemann, T. Gruber, G. Hager, D. Oryspayev, G. Wellein. "CloverLeaf on Intel multi-core CPUs: A case study in write-allocate evasion", *2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (San Francisco, CA, USA, 27–31 May 2024), IEEE, 2024, ISBN 9798350387124, pp. 350–360. ↑185

[302] *Fujitsu server performance report PRIMERGY TX2550 M7*, 2024 (Accessed 24.02.2025). ↑187

[303] *Fujitsu performance report PRIMERGY CX2550 M7 / CX2560 M7 v.1.3*, 2024 (Accessed 22.02.2025). ↑187, 188, 192, 198, 225

[304] *Memory performance of Xeon Scalable Processor (Sapphire Rapids / Emerald Rapids) based Systems v.1.1*, 2024 (Accessed 25.09.2024). ↑188

[305] B. Gray, D. Dumas, D. Shakshober, M. Mehta. *Red Hat Enterprise Linux performance results on 5th Gen Intel® Xeon® Scalable Processors*, 2024 (Accessed 22.12.2024). ↑189, 198

[306] *Fujitsu server performance report PRIMEQUEST 4400E v. 1.0*, 2023 (Accessed 22.12.2024). ↑190, 193

[307] O. Tatebe. *Data and computational science driven by Persistent Memory supercomputer Pegasus*, 2023 (Accessed 5.03.2025), 19 pp. ↑190

[308] M. Croci, G. N. Wells. *Mixed-precision finite element kernels and assembly: Rounding error analysis and hardware acceleration*, 2024, 37 pp. arXiv 2410.12614 ↑190

[309] *Using Intel's new built-in AI acceleration engines*, 2024 (Accessed 31.07.2024(*)), 57 pp. ↑190, 191

[310] H. Kim, G. Ye, N. Wang, A. Yazdanbakhsh, N. S. Kim. "Exploiting Intel® advanced matrix extensions (AMX) for large language model inference", *IEEE Computer Architecture Letters*, **23**:1 (2024), pp. 117–120. ↑190, 205

[311] E. Georganas, D. Kalamkar, K. Voronin, A. Kundu, A. Noack, H. Pabst. "Harnessing deep learning and HPC kernels via high-level loop and tensor abstractions on CPU architectures", *2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (San Francisco, CA, USA, 27–31 May 2024), IEEE, 2024, ISBN 9798350387124, pp. 950–963. ↑190, 211, 212

[312] *4th Generation Intel X Xeon Scalable Processor Benchmark Report*, https://axel.as-1.co.jp/images/pdf/hpcs/benchmark03.pdf, 2023 (Accessed 8.01.2025), 20 pp. (in Japanese). ↑

[313] V. Marjanović, J. Gracia, C. W. Glass. "Performance modeling of the HPCG benchmark", *High Performance Computing Systems. Performance Modeling, Benchmarking, and Simulation*, 5th International Workshop, PMBS 2014 (New Orleans, LA, USA, 16 November 2014), Lecture Notes in Computer Science, vol. **8966**, Springer, Cham, 2015, ISBN 978-3-319-17247-7, pp. 172–192. ↑193, 194

[314] S. Pareek, M. Naik, K. Veena. *16G PowerEdge platform BIOS characterization for HPC with Intel Sapphire Rapids*, 2023 (Accessed 8.01.2025). ↑194, 195

[315] M. Heroux, J. Dongarra, P. Luszczek. *HPCG technical specification*, Sandia Report SAND2013-8752, Sandia National Laboratories, 2013 (Accessed 8.03.2025), 21 pp. ↑195

[316] A. Fogli, B. Zhao, P. Pietzuch, M. Bandle, J. Giceva. "OLAP on modern chiplet-based Processors", *Proceedings of the VLDB Endowment*, **17**:11 (2024), pp. 3428–3441. ↑197

[317] J. Plana-Riu, F. Xavier Trias, A. Alsalti-Baldellou, G. Colomer, A. Oliva. "Performance analysis of parallel-in-time techniques in modern supercomputers", *ECCOMAS: 9th European Congress on Computational Methods in Applied Sciences and Engineering* (Lisbon, Portugal, 3–7 June 2024), International Centre for Numerical Methods in Engineering (CIMNE), 2024, ISBN 978-84-127483-6-9 (Accessed 20.03.2025), 10 pp. ↑198

[318] M. Larabel. *Ubuntu 24.04 + Linux 6.9 Intel & AMD server performance*, 2024 (Accessed 18.03.2025), 10 pp. ↑199, 200, 202, 230

[319] M. Larabel. *Intel 5th Gen Xeon performance benchmarks: impressive efficiency gains with "Optimized Power Mode"*, 2023 (Accessed 7.02.2025), 10 pp. ↑

[320] S. Pareek, K. Veena, M. Naik, P. Donthireddy. *HPC Application Performance on Dell PowerEdge C6620 with Intel 8480+ SPR*, 2023 (Accessed 30.11.2024). ↑199, 200, 203

[321] *NAMD GPU benchmarks and hardware recommendations*, 2024 (Accessed 24.03.2025). ↑202

[322] A. et al. Prabhakaran. *Scalable End-to-End Enterprise AI on 4th Gen Intel Xeon Scalable Processors*, White Paper, 2023 (Accessed 7.02.2025(*)). ↑204

[323] W. Huang, D. Wang, S. Zhou, C. Li, Y. Xie, P. He. "Accelerating deep learning workloads with advanced matrix extensions", *2023 5th International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)* (Hangzhou, China, 15–17 December 2023), IEEE, 2023, ISBN 9798350359923, pp. 70–73. ↑205

[324] J. V. de Oliveira Silva, T. Tahmid, W. da Silva Pereira. *Low precision and efficient programming languages for sustainable AI: Final report for the summer project of 2024*, NoNREL/TP-2C00-90910, National Renewable Energy Laboratory (NREL), Golden, CO, USA, 2024 (Accessed 10.05.2025), 27 pp. ↑205

[325] A. F. AbouElhamayed, J. Dotzel, Y. Akhauri, C.-C. Chang, S. Gobriel, noz J. Pablo Mu V. S. Chua, N. Jain, M. S. Abdelfattah. *SparAMX: Accelerating compressed LLMs token generation on AMX-powered CPUs*, 2025, 14 pp. arXiv 2502.12444 ↑205

[326] D. Quinn, M. Nouri, N. Patel, J. Salihu, A. Salemi, S. Lee, H. Zamani, M. Alian. "Accelerating retrieval-augmented generation", *ASPLOS '25: Proceedings of the 30th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*. V. 1 (Rotterdam, Netherlands, 30 March 2025–3 April 2025), ACM, New York, 2025, ISBN 979-8-4007-0698-1, pp. 15–32. ↑205

[327] S. Saha. *Taking on AI inferencing with 5th Gen Intel Xeon Scalable Processors*, 2024 (Accessed 2.01.2025). ↑205

[328] *OpenVINO 2024.6* (Accessed 17.01.2025). ↑205

[329] *Benchmark suite results. MLPerf inference: datacenter*, 2024 (Accessed 14.01.2025). ↑205, 206

[330] A. Eassa, A. Nanjappa, Zh. Jiang, Y. Zhang, J. Yang, Z. Kong, Sh. Xu. *NVIDIA Blackwell Platform Sets New LLM Inference Records in MLPerf Inference v4.1*, 2024 (Accessed 14.01.2025). ↑206

[331] T. Zhang, B. Patel, N. Sundararajan, J. Engh, Y. Qiu, L. Tsai, T. Iyengar, R. Chukka. *MLPerf™ Inference 4.0 on Dell PowerEdge Server with Intel® 5th Generation Xeon® CPU*, 2024 (Accessed 25.03.2025). ↑206

[332] T. Zhang, B. Patel. *Running LLMs on Dell PowerEdge Servers with Intel® 4th Generation Xeon® CPUs*, 2024 (Accessed 4.12.2025). ↑206

[333] V. G. Jithin, P. S. Ditto, M. S. Adarsh. *Inference acceleration for large language models on CPUs*, 2024, 9 pp. arXiv 2406.07553 ↑206

[334] A. Chu, A. Vance. *Achieve Leading AI Performance with 5th Gen Intel Xeon Processors and Open Source AI Software*, upd. 6/14/2024 (Accessed 2.01.2024(*)). ↑206, 207

[335] Pacheco F. Santana. *Performance evaluation of object detection in different architectures*, Master in High Performance Computing, SISSA, 2023 (Accessed 12.01.2025), 62 pp. ↑207, 217

[336] J. Terven, D. Cordova-Esparza. *A comprehensive review of YOLO: From YOLOv1 and beyond*, 2023, 36 pp. arXiv 2304.00501 ↑207

[337] *Performance Data for Intel AI Data Center Products* (Accessed 15.01.2025(*)). ↑209

[338] *Performance Data for Intel AI Data Center Products* (Accessed 15.01.2025(*)). ↑209

[339] M. Abuzaina, A. Bhuiyan, M. Minakshi. *Improve performance of TensorFlow AI Workloads Using 5th Gen Intel Xeon Scalable Processors*, 2023 (Accessed 15.01.2025(*)). ↑209, 210

[340] X. Jiang, M. Minakshi, R. Poornachandran, Sh. Najnin. "Power efficient deep learning acceleration using Intel Xeon Processors", *2024 IEEE High Performance Extreme Computing Conference (HPEC)* (Wakefield, MA, USA, 23–27 September 2024), IEEE, 2024, pp. 1–6. ↑210

[341] R. Hariharan. *Leadership XGBoost Performance Outperforminh 5th Gen Intel Xeon*, 2024 (Accessed 08.01.2026). ↑212

[342] T. Gamblin, M. LeGendre, M. R. Collette, G. L. Lee, A. Moody, B. R. de Supinski. "The Spack package manager: bringing order to HPC software chaos", *SC '15: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Austin, TX, USA, 15–20 November 2015), IEEE, 2015, ISBN 978-1-5090-0273-3, pp. 1–12. ↑213

[343] *NVIDIA HPC application performance*, 2024 (Accessed 9.10.2024). ↑216

[344] *NAMD GPU benchmarks and hardware recommendations*, 2024 (Accessed 12.01.2025). ↑216

[345] J. Linford. *Developing of Nvidia superchips*, 2024 (Accessed 8.01.2025), 73 pp. ↑216

[346] G. Olenik, M. Koch, Z. Boutanios, H. Anzt. "Towards a platform-portable linear algebra backend for OpenFOAM", *Meccanica*, **60** (2024), pp. 1659–1672. ↑217

[347] K. Halbiniak, K. Rojek, S. Iserte, R. Wyrzykowski. "Unleashing the potential of mixed precision in AI-accelerated CFD simulation on Intel CPU/GPU architectures", *24th International Conference on Computational Science* (Málaga, Spain, 2–4 July 2024), Lecture Notes in Computer Science, vol. **14837**, Springer, Cham, ISBN 978-3-031-63777-3, pp. 203–217. ↑217

[348] I. Sfiligoi, E. A. Belli, J. Candy. "The benefits of HBM memory for CPU-based fusion simulations", *PEARC '24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), ACM, New York, 2024, ISBN 979-8-4007-0419-2, id. 103, 2 pp. ↑217, 226

[349] J. E. Martin, C. Feldman, A. Calder, T. Curtis, E. Siegmann, D. Carlson, R. Gonzalez, D. Wood, R. Harrison, F. Coskun. "Benchmarking with Supernovae: A performance study of the FLASH code", *PEARC '24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), ACM, New York, 2024, ISBN 979-8-4007-0419-2, id. 8, 9 pp. ↑217

[350] V. Arslan. "HBM Contribution to Intel Sapphire Rapids for Geophysical Workloads", *EAGE Seventh High Performance Computing Workshop* (Lugano, Switzerland, 25–27 September 2023), vol. **2023**, European Association of Geoscientists & Engineers, 2023, pp. 1–5. ↑217

[351] Z. Li, G. Rickert, N. Zheng, Zh. Zhang, I. Özgen-Xian, D. Caviedes-Voullième. "SERGHEI v2.0: introducing a performance-portable, high-performance three-dimensional variably-saturated subsurface flow solver (SERGHEI-RE)", *Geoscientific Model Development*, **18**:2 (2025), pp. 547–562. ↑217

[352] N. Piroozan, N. Kumar. "Enabling performant thermal conductivity modeling with DeePMD and LAMMPS on CPUs", *Proceedings of the SC'23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), ACM, New York, 2023, ISBN 979-8-4007-0785-8, pp. 88–94. ↑217

[353] H. et al. Ibeid. *Performance Analysis of HPC applications on the Aurora Supercomputer: Exploring the Impact of HBM-Enabled Intel Xeon Max CPUs*, 2025, 11 pp. arXiv 2504.03632 ↑217, 218

[354] J. D. McCalpin. "Bandwidth limits in the Intel Xeon Max (Sapphire Rapids with HBM) processors", *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, May 21–25 2023), Lecture Notes in Computer Science, vol. **13999**, Springer, Cham, 2023, ISBN 978-3-031-40842-7, pp. 403–413 (Accessed 26.02.2025). ↑219, 220, 221, 229, 234, 245

[355] Y. Wang, J. D. McCalpin, J. Li, M. Cawood, J. Cazes, H. Chen, L. Koesterke, H. Liu, Ch.-Y. Lu, R. McLay, K. Milfield, A. Ruhela, D. Semeraro, W. Zhang. "Application performance analysis: a report on the impact of memory bandwidth", *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, May 21–25 2023), Lecture Notes in Computer Science, vol. **13999**, Springer, Cham, 2023, ISBN 978-3-031-40842-7, pp. 339–352 (Accessed 18.05.2025). ↑221, 226, 229

[356] R. Ristow Hadlich, G. Verma, T. Curtis, E. Siegmann, D. Assanis. "A64FX enables engine decarbonization using deep learning", *PEARC'24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), ACM, New York, 2024, ISBN 979-8-4007-0419-2, id. 52, 5 pp. ↑222

[357] N. A. Simakov, M. D. Jones, T. R. Furlani, E. Siegmann, R. J. Harrison. "First impressions of the NVIDIA Grace CPU Superchip and NVIDIA Grace Hopper Superchip for scientific workloads", *Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region Workshops* (Nagoya, Japan, 25–27 January 2024), ACM, New York, 2024, ISBN 979-8-4007-1652-2, pp. 36–44. ↑223, 224, 245

[358] M. Larabel. *Intel Xeon Max Enjoying some performance gains with Linux 6.6*, 2023 (Accessed 11.04.2025). ↑226

[359] M. Larabel. *Intel Xeon Max 9480/9468 show significant uplift in HPC & AI workloads with HBM2e*, 2023 (Accessed 9.04.2025). ↑226, 227, 228, 229

[360] C. Kutzner, V. Miletić, K. P. Rodríguez, M. Rampp, G. Hummer, B. L. de Groot, H. Grubmüller. "Scaling of the GROMACS molecular dynamics code to 65k CPU cores on an HPC cluster", *Journal of Computational Chemistry*, **46**:5 (2025), id. e70059. ↑226

[361] *Mdsv3 Very High Memory series*, 2024 (Accessed 13.04.2025). ↑228, 230

[362] *Dlsv5 sizes series*, 2024 (Accessed 13.04.2025) (in Russian). ↑230

[363] L. Britton, S. Malamed. *Public preview: new AMD-based VMs with increased performance, Azure Boost, and NVMe support*, 2023 (Accessed 6.10.2024). ↑230

[364] C. Yun. *New – Amazon EC2 Hpc7a instances powered by 4th Gen AMD EPYC™ processors optimized for high performance computing*, 2023 (Accessed 13.04.2025). ↑231

[365] *Developer clouds for accelerated computing* (Accessed 13.04.2025(*)). URL ↑231
[366] M. Thiagarajan. *Announcing World's Largest, first Zettascale AI supercomputer in the Cloud*, 2024 (Accessed 13.04.2025). URL ↑231
[367] T. P. Morgan. *Intel rounds out "Granite Rapids" Xeon 6 with a slew of chips*, 2025 (Accessed 17.05.2025). URL ↑231, 239
[368] M. D. Powell, P. Fleming, V. I. Krishna, N. Lakkakula, S. Ravisundar, P. Mosur. "Intel Xeon 6 product family", *IEEE Micro*, **45**:3 (2025), pp. 31–40. doi ↑231, 232
[369] C. Gianos. "Architecting for flexibility and value with Next Gen Intel® Xeon® processors", *2023 IEEE Hot Chips 35 Symposium (HCS)* (Palo Alto, CA, USA, 27–29 August 2023), IEEE, 2023, ISBN 9798350339086, pp. 1–15. doi ↑232
[370] *5th Gen AMD EPYC™ Processor Architecture 1st ed*, whitepaper, 2025 (Accessed 25.04.2025), 19 pp. URL ↑233
[371] *AMD EPYC™ 9005 Processor Architecture Overview*, whitepaper, 2025 (Accessed 25.04.2026), 11 pp. URL ↑233
[372] T. Singh, S. Oliver, S. Rangarajan, S. Southard, C. Henrion, A. Schaefer. ""Zen 5": The AMD high-performance 4nm x86-64 microprocessor core", *2025 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 16–20 February 2025), IEEE, 2025, pp. 1–3. doi ↑233
[373] B. Cohen, M. Subramony, M. Clark. "Next generation "Zen 5" core", *2024 IEEE Hot Chips 36 Symposium (HCS)* (Stanford, CA, USA, 25–27 August 2024), IEEE, 2024, pp. 1–27. doi ↑233, 234
[374] *Zen 5 – Microarchitectures – AMD* (Accessed 26.04.2025). URL ↑233
[375] M. Larabel. *Intel Xeon 6980P 1S performance with DDR5-6400/MRDIMM-8800* (Accessed 24.11.2024). URL ↑233
[376] *Intel® Xeon® 6*, Performance Index, 2025 (Accessed 26.04.2025(*)). URL ↑234
[377] R. et al. Sehgal. *Optimizing system memory bandwidth with Micron CXL memory expansion modules on Intel Xeon 6 processors*, 2024, 7 pp. arXiv 2412.12491 ↑234, 235
[378] M. Larabel. *AVX-512 performance with 256-bit vs. 512-bit data path for AMD EPYC™ 9005 CPUs*, 2024 (Accessed 12.10.2024). URL ↑234
[379] *OpenFOAM on 5th Gen AMD EPYC™ Processors*, 2024 (Accessed 1.12.2024). URL ↑235
[380] S. Siddiqi, R. Kern, M. Boehm. "SAGA: a scalable framework for optimizing data cleaning pipelines for machine learning applications", *Proceedings of the ACM on Management of Data*, **1**:3 (2023), id. 218, 26 pp. doi ↑240
[381] Y. Shen, X. Ren, Yu. Lu, H. Jiang, H. Xu, D. Peng, Y. Li, W. Zhang, B. Cui. "Rover: An online Spark SQL tuning service via generalized transfer learning", *KDD '23: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA, 6–10 August 2023), ACM, New York, 2023, ISBN 979-8-4007-0103-0, pp. 4800–4812. doi ↑240

[382] Y. Li, H. Jiang, Y. Shen, Y. Fang, X. Yang, D. Huang, X. Zhang, W. Zhang, C. Zhang, P. Chen, B. Cui. *Towards general and efficient online tuning for spark*, 2023, 14 pp. arXiv 2309.01901 ↑240

[383] J. Xin, K. Hwang, Z. Yu. "LOCAT: Low-overhead online configuration auto-tuning of Spark SQL applications", *SIGMOD '22: Proceedings of the 2022 International Conference on Management of Data* (Philadelphia, PA, USA, 12–17 June 2022), ACM, New York, 2022, ISBN 978-1-4503-9249-5, pp. 674–684. ↑240

[384] N. Perera, A. K. Sarker, M. Staylor, G. von Laszewski, K. Shan, S., Kamburugamuve Widanage C., Abeykoon V., Kanewela T. A., Fox G.. "In-depth analysis on parallel processing patterns for high-performance Dataframes", *Future Generation Computer Systems*, **149** (2023), pp. 250–264. ↑240

[385] U. E. Özdil, S. Ayvaz. "An experimental and comparative benchmark study examining resource utilization in managed Hadoop context", *Cluster Computing*, **26**:3 (2023), pp. 1891–1915. ↑240

[386] Y. Yuan, M. F. Salmi, Y. Huai, K. Wang, R. Lee, X. Zhang. "Spark-GPU: An accelerated in-memory data processing engine on clusters", *2016 IEEE International Conference on Big Data (Big Data)* (Washington, DC, USA, 05–08 December 2016), IEEE, 2016, ISBN 978-1-4673-9005-7, pp. 273–283. ↑

[387] P. Li, Y. Luo, N. Zhang, Y. Cao. "HeteroSpark: A heterogeneous CPU/GPU Spark platform for machine learning algorithms", *2015 IEEE International Conference on Networking, Architecture and Storage (NAS)*, IEEE, 2015, ISBN 9781467378901, pp. 347–348. ↑240

[388] R. Malakar, N. Vydyanathan. "A CUDA-enabled Hadoop cluster for fast distributed image processing", *2013 National Conference on Parallel Computing Technologies (PARCOMPTECH)* (Bangalore, India, 21–23 February 2013), IEEE, 2013, ISBN 9781479915903, pp. 1–5. ↑240

[389] A. Abbasi, F. Khunjush, R. Azimi. "A preliminary study of incorporating GPUs in the Hadoop framework", *The 16th CSI International Symposium on Computer Architecture and Digital Systems (CADS 2012)* (Shiraz, Fars, Iran, 02–03 May 2012), IEEE, 2012, ISBN 978-1-4673-1481-7, pp. 178–185. ↑240

[390] M. Azeem, B. M. Abualsoud, D. Priyadarshana. "Mobile Big Data analytics using deep learning and Apache Spark", *Mesopotamian Journal of Big Data*, **2023** (2023), pp. 16–28. ↑240

[391] R. Wu, Y. Wang, D. Kutscher. *Affordable HPC: leveraging small clusters for big data and graph computing*, 2024, 9 pp. arXiv 2408.15568 ↑240, 243

[392] R. Prichard, W. Strasser. "A Novel HPC scaling optimization methodology", *Proceeding of 7th Thermal and Fluids Engineering Conference (TFEC)* (Begellhouse, Las Vegas, NV, USA, 15–18 May, 2022), 2022, ISBN 978-1-56700-544-8, pp. 183–192. ↑240

[393] M. Rogowski, L. Dalcin, M. Parsani, D. E. Keyes. "Performance analysis of relaxation Runge–Kutta methods", *The International Journal of High Performance Computing Applications*, **36**:4 (2022), pp. 524–542. ↑241

[394] Y. Che, C. Xu, J. Fang, Y. Wang, Z. Wang. "Realistic performance characterization of CFD applications on Intel Many Integrated Core architecture", *The Computer Journal*, **58**:12 (2015), pp. 3279–3294. ↑241

[395] R. Prichard, W. Strasser. "When fewer cores is faster: a parametric study of undersubscription in high-performance computing", *Cluster Computing*, **27** (2024), pp. 9123–9136. ↑241

[396] S. Hammond, C. Vaughan, C. Hughes. "Evaluating the Intel Skylake Xeon processor for HPC workloads", *2018 International Conference on High Performance Computing & Simulation (HPCS)* (Orleans, France, 16–20 July, 2018), IEEE, 2018, ISBN 9781538678800, pp. 342–349. ↑241

[397] T. Zhang, Sh. Shirzad, B. Wagle, A. S. Lemoine, P. Diehl, H. Kaiser. *Supporting OpenMP 5.0 Tasks in hpxMP – a study of an OpenMP implementation within Task Based Runtime Systems*, 2020, 12 pp. arXiv 2002.07970 ↑241

[398] R. Prichard, W. Strasser. "An intuitive multi-objective, multi-variable high-performance computing optimization methodology", *International Journal of Computational Fluid Dynamics*, **38**:8–9 (2024), pp. 610–616. ↑242

[399] J. Chang, K. Lu, Y. Guo, Y. Wang, Zh. Zhao, L. Huang, H. Zhou, Y. Wang, F. Lei, B. Zhang. "A survey of compute nodes with 100 TFLOPS and beyond for supercomputers", *CCF Transactions on High Performance Computing*, **6**:3 (2024), pp. 243–262. ↑242, 243

[400] *HPC6, supercomputer at the service of energy* (Accessed 20.04.2025). ↑244

[401] *GPU nodes — LUMI-G* (Accessed 18.04.2025). ↑244

[402] *Perlmutter architecture* (Accessed 18.04.2025). ↑244

[403] *HPE Cray Supercomputing EX QuickSpecs v.19*, 2024 (Accessed 19.04.2025), 19 pp. ↑244, 245

[404] L. Fusco, M. Khalilov, M. Chrapek, G. Chukkapalli, Th. Schulthess, T. Hoefler. *Understanding data movement in tightly coupled heterogeneous systems: A case study with the Grace Hopper superchip*, 2024, 12 pp. arXiv 2408.11556 ↑245

[405] *AMD Instinct MI300A APU Datasheet*, 2023 (Accessed 21.01.2025), 2 pp. ↑245

(*) The access to this site is closed to Russian users (*editor's note*);

(**) This source comes unavailable to the journal's editors (*editor's note*).

Appendix. Abbreviations used in sections of the review

Here are given only some of the abbreviations that are not the most widely used in the relevant scientific literature (from the author's point of view), which are used in the text of the article in a wide variety of sections, and not only in those oriented towards concrete processor families.

Generic abbreviations:

1P one-processor *2.1, 2.2, 2.3, 2.4, 3.2, 4.1, 4.2, 4.3, 4.4, 6, 7*

2P two-processor *Introduction, 1.1, 1.3, 2.1, 2.2, 2.3, 2.4, 3, 3.1, 3.2, 4.1, 4.2, 4.3, 4.4, 6, 7*

1DPC One DIMM per channel *2.2, 2.4, 3.1, 3.3, 4.1*

2DPC Two DIMM per channel *2.2, 2.4, 3.1, 3.3, 4.1*

ACPI Advanced Configuration and Power Interface . *1.2, 3.2*

FMA Fused multiply-add *2.1, 2.4, 3*

NPB NAS Parallel Benchmarks *2.4, 4.1, 4.3, 4.4, 6*

PTS Phoronix Test Suite *Introduction, 2.4, 4.1, 4.2, 4.3, 4.4, 6, 7*

SKU Stock Keeping Unit *2.2, 2.4, 3.1, 3.2, 3.3, 4.1*

SMT Simultaneous multithreading . . . . . *Introduction, 2, 2.2, 2.3, 2.4, 3, 3.1, 4.1*

AMD Hardware Abbreviations:

CCD Core Complex Die *2.2, 2.3, 2.4, 3*

CCX Core Complex *2.1, 2.2, 2.4*

GMI Global Memory Interface *2.2*

IF Infinity Fabric *2.2, 2.3, 3*

IOD I/O Die (now also used in Xeon 6) *2.2, 2.3, 6, 7*

NPS Nodes Per Socket *2.2, 2.3, 2.4, 7*

xGMI eXternal Global Memory Interconnect *2.2, 2.3*

Abbreviations for Intel hardware:

1LM one-level memory *3.2*

2LM two-level memory *3.2*

AMX Advanced Matrix Extensions *1.3, 2.1, 3.1, 3.2, 3.3, 4.1, 4.2, 6, 7*

EMR Emerald Rapids . . . . . *Introduction, 2.4, 3, 3.1, 3.3, 4, 4.1, 4.2, 4.3, 4.4, 6, 7*

ICL Ice Lake . . . *Introduction, 1.1, 2.1, 2.4, 3, 3.1, 4.1, 4.3, 4.4*

OPM Optimized Power Mode *3.1, 4.1, 4.2*

SNC (for SNC1, SNC2, SNC4) sub-NUMA clustering *3, 3.1, 3.2, 3.3, 4.1, 4.4*

SPR Sapphire Rapids . . . . . *Introduction, 1.1, 1.2, 1.3, 2.1, 2.4, 3, 3.1, 3.2, 3.3, 4, 4.1, 4.2, 4.3, 4.4, 6, 7*

UPI Ultra Path Interconnect *3, 3.1, 3.2, 3.3, 6*

Analogues of the terms Intel and AMD:

AMD	*Intel*
NPS	SNC
xGMI	UPI

Received 31.05.2025;
approved after reviewing 14.10.2023;
accepted for publication 10.11.2025;
published online 15.01.2026.

Recommended by *prof. S.M. Abramov*

Information about the author:

Mikhail Borisovich Kuzminsky

Senior Researcher, Laboratory of Computer Software for Chemical Research, Candidate of Chemical Sciences, Institute of Organic Chemistry, Russian Academy of Sciences. The scientific interests are high-performance computing, computer hardware, computational chemistry.

0000-0002-3944-8203

e-mail: *kus@free.net*

The author declare no conflicts of interests.

RU EN ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ ISSN 2079-3316
Обзорная статья ПРОГРАММНОЕ И АППАРАТНОЕ ОБЕСПЕЧЕНИЕ РАСПРЕДЕЛЕННЫХ И СУПЕРКОМПЬЮТЕРНЫХ СИСТЕМ

УДК 004.051+004.272+004.318+004.382+004.8+004.9
10.25209/2079-3316-2025-16-5-43-514

Вокруг условного 4-го поколения современных серверных процессоров AMD и Intel: их микроархитектура и производительность соответствующих вычислительных систем

Михаил Борисович **Кузьминский**✉
Институт органической химии им. Н. Д. Зелинского РАН, Москва, Россия
✉*kus@free.net*

Аннотация. Обзор посвящен особенностям микроархитектуры и производительности процессоров Intel Xeon – масштабируемых процессоров 4-го поколения (с микроархитектурой Sapphire Rapids-SP, далее Xeon SPR), 5-го поколения (Emerald Rapids-SP, далее Xeon EMR), и разных классов процессоров AMD EPYC архитектуры Zen 4, а также вычислительным системам на их основе. Анализируются данные о моделях Xeon SPR (и Xeon SPR с памятью HBM, то есть Xeon Max), Xeon EMR и процессорах AMD EPYC 9004 (хотя приведены и краткие данные о EPYC 8004 и 4004).

Эти процессоры отнесены в обзоре к условному 4-му поколению Xeon и EPYC. Сопоставления проводятся и с масштабируемыми процессорами Xeon 3-го поколения – Ice Lake-SP (далее Xeon ICL), Cooper Lake-SP, с AMD EPYC с архитектурой Zen 3 (Milan), а также иногда с процессорами ARM-архитектуры и GPU.

Кратко обсуждаются средства разработки программ (SDK) для процессоров 4-го поколения, имеющие важное значение для достигаемой производительности. В связи с применением чиплетов или использованием HBM-памяти в рассматриваемых процессорах AMD и Intel особое внимание обращается на поддерживаемые варианты NUMA.

Анализируется также аппаратная поддержка средств обеспечения безопасности для задач виртуализации, которые теперь часто применяются и в области высокопроизводительных вычислений (HPC).

Данные о производительности в обзоре охватывают широкий спектр областей применения, характерных для серверов с этими процессорами. Но основное внимание уделяется HPC и, в меньшей степени, задачам ИИ.

Рассматриваемые процессоры анализируются с точки зрения построения с ними гомогенных или содержащих GPU гетерогенных серверов и вычислительных систем на их основе (кластеров и суперкомпьютеров).

Анализируется также начальная информация о новейших процессорах Intel Xeon 6 Granite Rapids и AMD EPYC Zen 5 Turin, включая первые данные об их производительности.

Сделаны выводы общего характера о состоянии и образовавшихся тенденциях развития таких процессоров x86. (*Связанные тексты статьи на английском и на русском языках*)

Ключевые слова и фразы: x86, Zen 4, Genoa, Bergamo, Zen 5,Turin, Sapphire Rapids, Xeon Max, Emerald Rapids, Xeon 6, Granite Rapids, микроархитектура, производительность, HPC, ИИ, суперкомпьютеры

Для цитирования: Кузьминский М. Б. *Вокруг условного 4-го поколения современных серверных процессоров AMD и Intel: их микроархитектура и производительность соответствующих вычислительных систем* // Программные системы: теория и приложения. 2025. **Т. 16**. № 5(68). С. 43–514. (*Англ.+русс.*) https://psta.psiras.ru/read/psta2025_5_43-514.pdf

Введение

Современные процессоры x86-64 от Intel и AMD по-прежнему остаются наиболее распространенными серверными процессорами в мире для самых разных областей применения. В области суперкомпьютеров это легко увидеть из списка TOP500 [1]. Это же имеет место и для всех областей высокопроизводительных вычислений (HPC) и ИИ, хотя для таких задач процессоры x86-64 очень активно используются также в гетерогенных серверах совместно с GPU [2]. Среди тех областей, в которых GPU по-прежнему не применяются, указываются, например, и задачи ядерной безопасности [3].

Для уточнения сравнительных показателей производительности таких процессоров и GPU полезно сразу дать иллюстрацию. Например, в [4] показано, что на двухпроцессорной (2P-) системе с Intel Xeon Platinum 8480+ при умножении квадратных матриц разных размеров достигается производительность, очень близкая к получаемой на одном GPU Nvidia A100 без использования тензорных ядер, но энергопотребление GPU гораздо ниже.

Но современные суперкомпьютеры часто строятся и на гомогенных узлах без применения ускорителей. Из 50 самых мощных суперкомпьютеров ноябрьского списка TOP500 2023 года [5] на гомогенных серверах с процессорами x86-64 от Intel и AMD построено 22%, в том числе с применением новейших процессоров четвертого поколения, рассматриваемых в данном обзоре. Хотя появление суперкомпьютеров с новейшими GPU уменьшило эту долю в июньском списке TOP500 2024 года до 14%, в TOP500 появляются и новые суперкомпьютеры с гомогенными узлами.

В обзоре анализируются только серверные процессоры x86-64, они будут далее для краткости именоваться просто x86. Под поколениями процессоров x86 далее подразумеваются поколения AMD EPYC Zen-архитектур и масштабируемых процессоров Xeon.

В частности, AMD EPYC Zen 4 и масштабируемые процессоры Xeon четвертого и пятого поколения (Xeon SPR и Xeon EMR), которым далее уделяется основное внимание, отнесены к условному четвертому поколению x86.

Лидирующими по пиковой производительности (14 TFLOPS для FP64) в мире сегодня стали китайские RISC-процессоры SW39000 [6], но они ориентированы в основном на суперкомпьютер Sunway.

В мире расширяется применение серверных ARM-процессоров, но они не достигли уровня однозначного опережения x86 по производительности, что имело место и при появлении Fujitsu A64FX [7]. В основном ARM конкурируют с x86 по энергоэффективности (отношению производительности к энергопотреблению) и при использовании облачной технологии (см., например, [8–10]), хотя успехи ARM наблюдались и в отношении стоимость/производительность (см., например, [11]).

Применение новых серверных ARM-процессоров в HPC могло бы начать расширяться за счет использования высокопроизводительных ARM-процессоров на базе архитектуры Neoverse V2, но кроме 72-ядерного Nvidia Grace в гетерогенных системах с GPU Nvidia H100 и H200 (см., например, данные [2, 12]) к моменту написания обзора известно только о 96-ядерном Amazon AWS Graviton 4, а данные, демонстрирующие их существенные успехи в производительности по сравнению с x86, практически отсутствуют.

Интересное сопоставление микроархитектуры ARM Neoverse V2 в Nvidia Grace Superchip и рассматриваемых в обзоре процессоров x86 проведено в [13], где отмечается соревнование Nvidia, Intel и AMD в гонке за лучший процессор. Однако сопоставление процессоров четвертого поколения Intel и AMD с ARM в практическом плане интересно на основе реальной производительности, а не на основе собственно микроархитектуры, и данные в том числе и из [13] говорят о четкой конкурентоспособности Grace в основном для серверов с GPU, где важна более высокая пропускная способность памяти для Nvidia Grace – за счет применения более дорогой памяти LPDDR5X (если не учитывать Xeon Max с еще более дорогой HBM).

Важной также бывает задача оптимального выбора процессора для специфических приложений [14].

Лидерство процессоров x86 по производительности относительно других процессоров не имеет отношения к преимуществам или недостаткам CISC по сравнению с RISC, поскольку аппаратура x86 давно переводит свои команды с ISA x86 в микрооперации RISC-архитектуры, которые и выполняются [15, 16]. Этот факт можно рассматривать как подтверждение преимущества RISC, как и недавнее APX-расширение ISA Intel [17], где будет применяться скорее характерное для RISC большее число регистров общего назначения и команды, работающие с тремя регистрами.

Но реально на внеочередную (OoO, Out-of-Order) обработку команд сегодня требуется весьма много аппаратуры в каждом ядре процессора. Для OoO нужна аппаратная поддержка планирования исполнения последующих команд на свободных исполнительных блоках, предсказание переходов и другие компоненты фронт-энд части современных суперскалярных процессоров, дающие возможность одновременно выполнять по мере доступности исполнительных блоков максимально возможное число команд. На это уходит больше ресурсов, чем требуется из-за различий CISC и RISC.

Появившиеся очень давно дискуссии на тему RISC vs CISC активизировались во время появления ARM [18–20], и поныне продолжаются и в Internet. О фактической «пренебрежимости» преимуществами RISC или CISC говорилось уже давно [20]. По мнению автора, для задач

максимизации производительности в настоящее время более справедливой является точка зрения о пренебрежимости отличия между RISC и CISC-архитектурами по сравнению с используемыми современными сложными реализациями фронт-энд части соответствующих суперскалярных микропроцессоров (см., например, [21]). Важнее могут оказаться отличия собственно исполнительных устройств разных архитектур процессоров.

Преимущества современных моделей x86 сегодня связаны с их реализацией на уровне микроархитектуры, высоким уровнем используемой в них полупроводниковой технологии и массовым объемом поставок, дающих финансовые выгоды. Интегрально главное, на что ориентируется, например, AMD с точки зрения фронт-энд части ядра – это рост величины IPC (instructions per clock). В докладах AMD [22] или современных документах AMD по Zen 4 [23] на фронт-энд вообще обращается довольно мало внимания: изменения в исполнительных устройствах, высшие уровни иерархии памяти, и межсоединения ядер и процессоров важнее. В обзоре фронт-энд части процессоров x86 также будет рассматриваться относительно коротко.

Самым ярким показателем современных семейств серверных процессоров пожалуй является большое количество используемых там ядер. В Xeon SPR их минимум 8. В Xeon EMR их не меньше 16, как и в EPYC Zen 4 серии 9004. Для ряда задачи и приложений не требуется такого количества ядер. На современных серверах, кроме обыкновенной возможности одновременного выполнения несколько приложений в рамках одной ОС, широко применяются возможности виртуализации, для чего в процессорах используются и специальные аппаратные усовершенствования. Виртуализация и облачные технологии стали активно применяться и для задач HPC. Согласно [24], в не менее чем 48 суперкомпьютерах, занимающих места с 1 по 172 в TOP500 (июнь 2022 г.), используют контейнеры.

Обзор наиболее ориентирован на HPC (включая и суперкомпьютерный уровень), в меньшей степени – на ИИ, и еще меньше – на виртуализацию и облачные технологии.

Преимущества AMD EPYC по производительности в области виртуализации и применения облачных технологий (для массового применения не в области HPC и ИИ), в том числе и над Intel Xeon связаны с большим числом доступных в процессорах ядер и сформировались уже лет пять назад. Данных, демонстрирующих преимущества производительности EPYC Zen 2 и Zen 3, достаточно много. Пример для HPC – работы в области вычислительной гидродинамики (CFD) [25].

Много соответствующих данных для массово используемых областей применения этих серверных процессоров можно найти, например, в результатах разных OSG (Open System Group)-тестов SPEC, включая и SPEC CPU 2017 [26], или в данных сайта OpenBenchmarking [27] для тестов Phoronix Test Suite (PTS) [28]. Преимущества вышеуказанных серий EPYC в этих данных можно увидеть, например, сопоставляя производительности старших моделей этих поколений процессоров AMD с аналогичными моделями Xeon в соответствующие времена, или учитывая отношение стоимость/производительность.

Это подтверждается рядом соответствующих публикаций на сайте Phoronix (см., например, [29–31]). Исключения в основном относятся к случаям с активным использованием AVX-512. Небольшая информация с сопоставлениями производительности с Xeon третьего поколения (Xeon ICL и Xeon Cooper Lake-SP) и EPYC Zen 3 (Milan) приводится далее для демонстрации тенденций развития в четвертым поколении x86 от Intel (включая масштабируемые процессоры Xeon пятого поколения) и AMD, рассмотрению которых посвящен данный обзор.

Учитывая большое количество доступных данных о производительности для разных областей применения (в том числе указанных выше), не относящиеся к областям HPC и ИИ, соответствующие данные об улучшении производительности рассматриваемых в обзоре EPYC с архитектурой Zen 4 и масштабируемых процессоров Xeon четвертого и пятого поколения относительно их предыдущих поколений иллюстрируются дальше гораздо более ограничено.

К кратко рассматриваемым в обзоре областям относятся и активно анализируемые в последние времена проблемы возможного нарушения безопасности. Внимание к вопросам безопасности в последнее время возросло и в связи с ростом числа ядер в процессорах, сопровождавшимся использованием средств виртуализации в пределах уже одного процессора.

Проблемы безопасности имеют место для различных интегральных схем [32]. Архитектура фон Неймана уязвима для возможных нарушений безопасности (см., например, [33]). В суперскалярных процессорах, поддерживающих SMT, OoO и предсказание переходов, возможности нарушения безопасности возрастают и реализуемы на процессорах самых разных архитектур – x86, ARM, RISC-V [34, 35]. Естественно, в первую очередь такая информация появляется для x86 из-за гораздо большей распространенности (см., например, [36, 37]).

В четвертом поколении серверных процессоров Intel и AMD имеются специальные аппаратные средства для упразднения определенных возможностей нарушения безопасности. В последние времена появляется много публикаций о проблемах безопасности процессоров и проводятся подробные исследования в том числе и этих новых средств (например, для Intel – [38]).

Но эти средства в первую очередь относятся к возможным нарушениям безопасности, возникающим при использовании средств виртуализации. Возможно, эти аппаратные доработки не могут подавить все возможные нарушения безопасности ни в процессорах Intel, ни в процессорах AMD, поскольку появляются данные о новых типах атак.

Для разных поколений архитектур AMD Zen и Intel Xeon, включая и рассматриваемое в обзоре четвертое поколение, небезопасность виртуализации демонстрируется в [**39**–**44**]. Среди этих публикаций большее число посвящено атакам разных поколений AMD Zen. Соответствующие доработки в процессорах способствуют в первую очередь затруднению возможных нарушений безопасности и уменьшению вероятности их реализации. Аппаратные средства повышения безопасности рассматриваются далее в разделах про микроархитектуры конкретных семейств процессоров в первую очередь для задач виртуализации и соответственно облачной технологии.

Современные процессоры x86 от Intel сопровождаются еще рядом новых интересных аппаратных усовершенствований, реализованных в виде акселераторов. Они являются специализированными и полезны для некоторых важных, но узко ориентированных областей их применения и поэтому в обзоре рассматриваются достаточно кратко (см. о них далее в разделе 3).

Далее в разделе 1 анализируются важные общие аппаратные и программные возможности, в том числе сформировавшиеся до возникновения процессоров условного четвертого поколения. В разделе 2 рассматривается архитектура процессоров Zen 4 и вычислительные системы на их основе, а также данные об их производительности. В разделе 3 анализируется информация об архитектуре различных процессоров Xeon, отнесенных здесь к условному четвертому поколению, и вычислительных систем на их основе. Поскольку в разделе 3 объединены данные для разных процессоров Xeon, обсуждение производительности систем на их базе, в том числе в сравнении с Zen 4, проводится в отдельном разделе 4. В небольшом разделе 5 рассматривается информация о применении вычислительных систем на базе x86 четвертого поколения для виртуализации и облачных технологий. В разделе 6 анализируются данные о новейших высокопроизводительных процессорах Xeon 6 и Zen 5, включая первоначальные данные о производительности. Раздел 7 посвящен посвящен построению с применением x86 гомогенных и гетерогенных узлов кластеров или суперкомпьютеров.

В приложении дается список широко используемых в обзоре сокращений, которые, с нашей точки зрения, не распространены в литературе.

1. Общее для рассматриваемых процессоров x86, в том числе сложившееся до появления их четвертого поколения

1.1. Об аппаратных средствах x86 предыдущих поколений

Из-за быстрого прогресса технологии, используемой для производства процессоров, в них интегрируется больше функций. Отпала необходимость применения отдельного южного моста на материнской плате. Поскольку активная ориентация на SoC продолжается, с точки зрения автора в ближайшем будущем в современных серверных процессорах AMD и Intel вероятно исчезнет и применение чипсета. Для процессоров AMD он уже не используется, хотя масштабируемые процессоры Intel Xeon третьего и четвертого/пятого поколения еще имеют чипсет (C620A и C741 соответственно).

Наиболее важные для процессоров функции перестают базироваться на чипсете. Если в чипсете C620A имелись еще аппаратные средства акселераторов, то C741 их уже не содержит (они интегрированы в Xeon). Чипсет C741 поддерживает работу с SATA, USB, некоторые функции безопасности и др. [45]. Современные процессоры EPYC имеют интегрированный блок, отвечающий за все потребности ввода и вывода данных (имеются в виду не просто ввод-вывод, а любые передачи данных к и от процессора, включая и работу с оперативной памятью). Подробнее см. далее в разделе 2.2.

Во многих современных серверных процессорах от разных разработчиков используется многокристальный подход с чиплетами с двумерной (2D-) технологией. Применение многокристального подхода способствует сокращению количества отходов в производственном процессе. При размещении на кремниевой пластине монолитных процессоров один дефект может вызвать неработоспособность всего процессора, а при многокристальном подходе дефект относится только к одному кристаллу, дефектный кристалл просто отбраковывается, и в общем корпусе собираются только кристаллы без дефектов.

AMD одна из первых крупнейших компьютерных фирм мира, которая стала применять такую технологию [46] и использует этот подход также в своих GPU [2]. В результате сокращения количества отходов AMD получает возможность предлагать не самые высокие цены на свою продукцию.

Intel также очень давно стала использовать многокристальные архитектуры. В масштабируемых процессорах Xeon Intel стала применять многокристальные технологии, начиная со второго поколения [47]. Однако из-за технологических ограничений и меньшего количества применяемых в серверных процессорах ядер Intel до сих пор использовала этот подход в более ограниченной степени.

То, что было типичным в области производительности x86 (в первую очередь для HPC и ИИ) перед появлением 4-го поколения AMD Zen и Intel Xeon Scalable, мы опишем только «в целом», на макроуровне, поскольку сам анализ данных производительности для соответствующих поколений x86 к тематике обзора не относится.

AMD на рынке процессоров давно отличалась от Intel более низкой стоимостью своих аппаратных средств, аналогичных выпускаемым Intel [48]. Некоторые сравнительные данные для Zen 3 и масштабируемых процессоров Xeon 3-го поколения представлены в таблице 1.

Таблица 1. Краткая сопоставительная информация о третьем поколении x86 от Intel и AMD

	Число ядер в CPU	Частота, ГГц	Цена[1] одного CPU	Число CPU	SPEC CPU2017 fp_speed (base/peak)[2]	SPEC CPU2017 fp_rate (base/peak)[2]
EPYC 7763	64	2.4–3.75	$7890	2	263/270[4]	663/7101[3]
				1	177/181[5]	333/3571[3]
EPYC 7663	56	2–3.5	$6366	2	246/253[6]	596/637[4]
				1	162/162[6]	299/3201[3]
EPYC 7643	48	2.3–3.6	$4995	2	236/245[6]	576/617[6]
				1	153/158[7]	288/3041[4]
EPYC 7543	32	2.8–3.7	$3761	2	232/242[4]	531/540[4]
				1	153/158[7]	260/268[6]
EPYC 75F3	32	2.95–4	$4860	2	238/248[4]	546/565[4]
				1	160/165[5]	273/2831[3]
Xeon 8380HL	28	2.9–4.3	$13012	4	255/255[8]	667/7061[5]
Xeon 8380	40	2.3–3.4	$8978[3]	2	277/277[9]	605/640[10]
Xeon 8368	38	2.4–3.4	$7214	2	269/260[10]	586/620[10]
Xeon 8362	32	2.8–3.6	$6236	2	270/270[11]	561/589[11]
Xeon 8358	32	2.6–3.4	$4607	2	251/251[12]	507/534[10]

[1] Данные на 12.05.2024 для AMD – из [49], для Intel – из [50]
[2] Данные из [51] с максимальной представленной там величиной base.
[3] Last Price at 2022-04-03 (ранее цена была выше) из [52]. Данные от 12.05.2024
Направители результата и серверы, на которых получены эти результаты:
[4] ASUS RS720A-E11 (KMPP-D32);
[5] ASUS RS520A-E11(Z12PP-D32);
[6] Cisco UCS C255 M6;
[7] Lenovo ThinkSystem SR655;
[8] Lenovo ThinkSystem SR860;
[9] xFusion 2288H V6 и ASUS RS720-E10-RS12(Z12PP-D32);
[10] ASUS RS720-E10-RS12(Z12PP-D32);
[11] ASUS RS720-E10(Z12PP-D32);
[12] Dell PowerEdge R650;
[13] ASUS RS520A-E11(KMPA-U16);
[14] Dell PowerEdge R6515;
[15] New H3C UniServer R6900 G5.

Примечание. Для уточнения данных об использовавшихся серверах в круглых скобках иногда указана применявшаяся в тесте материнская плата.

Очень важной общей особенностью рассматриваемых в обзоре процессоров Intel и AMD, на которую обращается особое внимание в обзоре, является устремление обоих фирм к повышению пропускной способности используемой памяти. Производительность микропроцессоров растет

быстрее пропускной способности памяти [53], которая может лимитировать производительность приложений. Про это у процессоров известно еще с прошлого века [54]. Поскольку в мире x86-процессоры AMD и Intel до сих пор используются в серверах чаще других, тут влияние пропускной способности памяти проявляется наиболее широко.

Процессоры и ядра Xeon ранее обычно опережали соответствующие аппаратные средства AMD по производительности. Предыдущие успехи AMD с серверными процессорами Opteron были во многом связаны с высокими показателями пропускной способности памяти [55]. Наблюдаемые сегодня успехи AMD Zen разных поколений также сочетались с данными о более высокой пропускной способности памяти для одного ядра по сравнению с ядрами Intel [56]. Хотя эти данные относятся только к чтению из памяти, они дают разумную оценку пропускной способности на ядро [56]. Информация о пропускной способности памяти для рассматриваемых в обзоре процессоров будет обсуждаться далее.

Далее в тексте для ситуаций, когда производительность ограничивается пропускной способностью памяти, применяется словосочетание «связанные памятью». Для массово используемых тестов производительности и приложений, не применяющих активно умножения матриц и AVX-512, особенно связанных памятью (например, для задач CFD) во втором и третьем поколениях EPYC Zen сформировались преимущества AMD по производительности. Кроме большего числа ядер они связаны, в частности, с указанной выше пониженной пропускной способностью оперативной памяти на одно ядро в Xeon соответствующих поколений по сравнению с AMD [56].

В интернете давно появляются сообщения о предпочтительности серверных процессоров AMD EPYC по сравнению с Intel Xeon. Ссылки на соответствующие этому данные о преимуществах по производительности процессоров AMD в массово используемых тестах производительности и приложениях приведены выше во введении. Дополнительной общей иллюстрацией являются и данные таблицы 1. Для примера сравнительных оценок стоимости и производительности укажем еще на [52].

Отметим, что в анализе производительности немаловажно также учитывать и даты получения соответствующих результатов (что связано с версиями ставших доступными и использованных SDK). Так, в таблице 1 Xeon Platinum 8380 немного опередил EPYC 7763 в тесте SPECcpu2017 fp_speed. Эти данные в таблице для Xeon 8380 (как и в тесте SPEC CPU2017 fp_rate) получены в 2023 году, а для EPYC 7763 – в 2021 году. Максимальный результат Xeon 8380 в 2021 году был 241/244 для base и peak-вариантов теста fp_speed, то есть ниже, чем у EPYC 7763.

В качестве другого примера укажем данные о производительности от отдельных ядер до серверов (в которых применялись в том числе Xeon ICL и AMD EPYC Zen 3, Milan) и небольших кластеров в наборе задач вычислительной химии с применением широко используемых в мире приложений. Эти данные обычно показывают преимущество ряда моделей EPYC Milan над старшими моделями Xeon ICL [57].

Особенно полезные для HPC данные [8] отличаются широким охватом разных областей. Часть этих данных представлена в таблице 2.

Таблица 2. Сопоставление производительности 2P-серверов на базе Xeon ICL и EPYC Zen 3

Производительность	Xeon 8380 (40 ядер)	EPYC 7763 (64 ядра)
DGEMM (GFLOPS)	**3824**	3046
DGEMM на ядро (GFLOPS)	**47.8**	23.8
HPL (GFLOPS)	1713	**2176**
HPL на ядро (GFLOPS)	**21.4**	17.0
FFT (GFLOPS)	**76.4**	54.7
FFT на ядро (GFLOPS)	**0.96**	0.43
GROMACS (нс/день)	133.3	**169.9**
NWChem (секунд)	**19.2**	26.7
OpenFOAM (минут)	6.8	**6.6**

Данные из [8].
Жирным шрифтом помечены более высокопроизводительные результаты.

Эти данные четко демонстрируют сложившееся в третьем поколении процессоров x86 от AMD и Intel: преимущество последней благодаря применению AVX-512 в DGEMM, которое в HPL (High Performance Linpack) уже элиминируется из-за большего числа ядер в AMD Zen 3. В таблице 2 кроме данных о трех проведенных тестов из HPCC [58] (DGEMM, HPL, FFT–FFTW3[1]) приведены данные тестов известных высокоэффективным распараллеливанием приложений вычислительной химии – молекулярной динамики (GROMACS с системой из примерно 82 тысяч атомов) и квантовой химии (NWChem, с маленькой системой – ион Au+, методом ХФ с последующим MP2 и связанными кластерами).

Еще один приведенный тест, средств OpenFOAM (Open Source Field Operation And Manipulation CFD ToolBox) относится к области механики сплошных сред (motorBike – расчет несжимаемого потока воздуха вокруг мотоцикла). Хотя OpenFOAM – это набор инструментов на C++ для решения в дискретной форме систем уравнений в частных производных в пространстве и времени, и может применяться не только в области механики сплошных сред. Но изначально он был направлен на CFD, и поэтому далее в обзоре рассматривается совместно с CFD-приложениями.

[1] https://www.fftw.org/, accessed 23.11.2024.

Сравнение с ARM-процессорами в [8] показало преимущества x86 в производительности; ARM могут быть впереди по энергоэффективности. Относительная производительность сильно зависит от приложений: GROMACS и OpenFOAM были быстрее в EPYC, а NWChem – в Xeon ICL. Соответственно надо внимательно смотреть, о данных каких тестов производительности идет речь.

Но в целом можно сказать, что и в приводимых здесь данных EPYC Zen 3 явно чаще оказывались более высокопроизводительными, чем Xeon ICL.

В качестве примера общих взвешенных рекомендаций для HPC и ИИ по выбору процессоров Xeon и EPYC третьего поколения с учетом и отношения цена/производительность укажем на данные сайта Microway [59, 60]. Для иллюстрации приведем здесь также пиковые производительности для разных моделей Xeon ICL – рисунок 1, рисунок 2 и для Zen 3 (Milan) – рисунок 3.

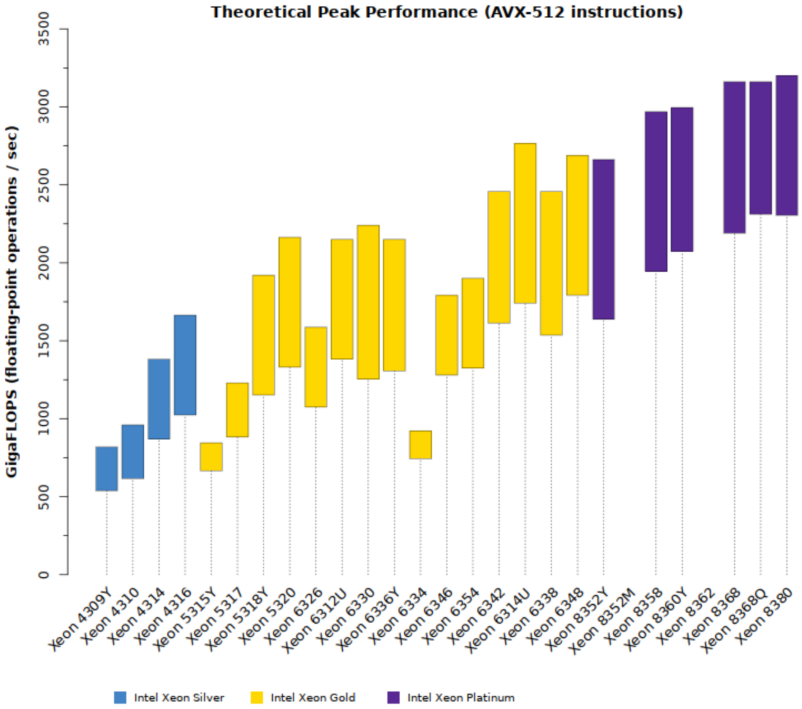


Рисунок 1. Пиковая производительность разных моделей Xeon ICL при использовании AVX-512 (рисунок из [59])

На рисунке 1 данные о пиковой производительности для FP64 в разных моделях Xeon ICL относятся к использованию AVX-512, а на рисунке 2 -к работе только с AVX2. На рисунке 3, для Zen 3, пиковые производительности разных моделей EPYC (там поддерживается только AVX2) рассчитаны для базовых тактовых частот, а теоретические данные для турбо-частот отображаются вершинами пунктирных линий.

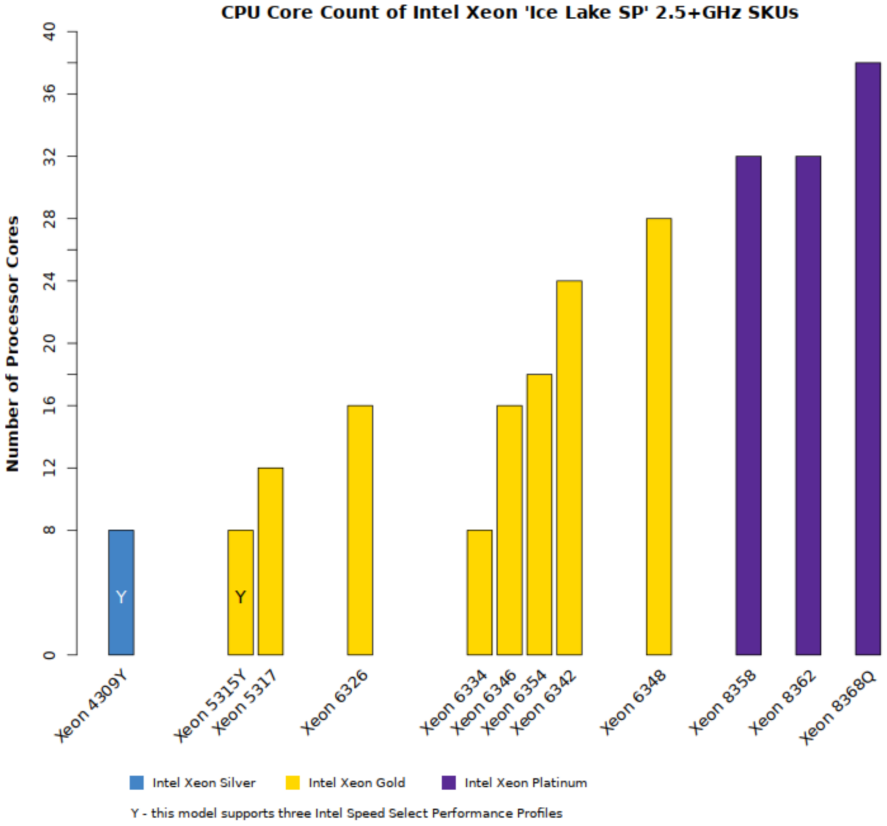


Рисунок 2. Пиковая производительность разных моделей Xeon ICL при использовании только AVX2 (рисунок из [59])

Несмотря на указанные выше преимущества в производительности процессоров EPYC предыдущих поколений Zen-архитектуры, реально по активности применения x86 на суперкомпьютерах в июньской версии TOP500 2024 года [1] предпочтительности использования x86 от AMD в первых 50 позициях данного списка не видно: EPYC применялись в 22 системах, Xeon — в 27. Статистические данные по всему списку [61] говорят

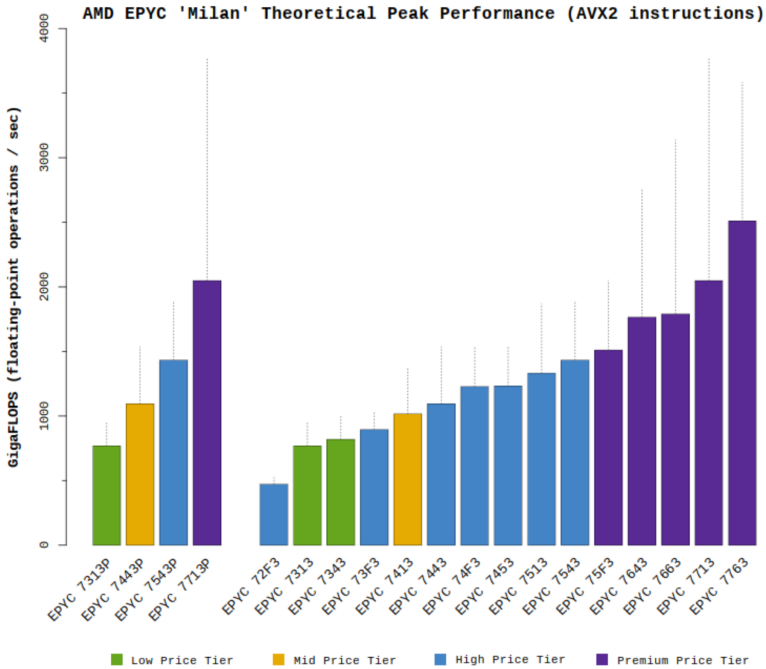


Рисунок 3. Пиковая производительность разных моделей EPYC Milan (рисунок из [60])

о применении второго поколения масштабируемых процессоров Xeon (Cascade Lake) в 123 суперкомпьютерах из TOP500, а второго поколения EPYC Zen 2 – в 68 суперкомпьютерах. Процессоры третьего поколения Xeon ICL применялись в 42 суперкомпьютерах этого списка, а EPYC Zen 3 Milan – в 73 суперкомпьютерах. Наконец, процессоры четвертого поколения Xeon SPR применялись там в 31 суперкомпьютере, а EPYC 9004 – в 15.

Также интересно отметить, что из 123 суперкомпьютеров с Xeon Cascade Lake в 79 использовались Xeon Gold 62xx, а более старшие серии Xeon Platinum 82xx и 92xx применялись на раза в два меньшем числе суперкомпьютеров. Задачи отбора моделей x86 для построения кластеров и суперкомпьютеров обсуждаются далее в отдельном разделе 7.

Обзор производительности ориентирован далее в основном на четвертое и пятое поколения масштабируемых процессоров Xeon и четвертое поколение Zen (EPYC) серии 9004 – как современных, ставших доступными примерно в одно и то же время, у которых есть достаточно много данных о достигаемой производительности. Ниже они объединены

в одно общее условно четвертое поколение серверных процессоров x86. Intel и AMD, естественно, приводят сопоставительные данные о росте производительности этих процессоров относительно их предыдущих поколений.

Соответственно сравнения производительности будут включать и данные для третьего поколения серверных процессоров (AMD Milan и Intel Xeon ICL). В разделе 6 анализируются и первоначальные данные о следующем поколении x86 от Intel и AMD – Xeon 6 (Granite Rapids) и Zen 5, включая их производительность. Но базовыми мы считаем сопоставления производительности условного четвертого поколения, поэтому в заголовке статьи и указано «вокруг четвертого поколения».

1.2. О регулировке частоты процессоров и использовании этого в Linux

Вопросы энергопотребления приобретают в последнее время все большее значение, и для вычислительных систем это может выражаться в двух аспектах: оптимизация отношения стоимость/производительность или рассмотрение потребляемой энергии как ограничения. Это в программном и аппаратном плане, включая современные серверные процессоры x86, анализируется в [**62**]. Здесь рассматривается только один важный, имеющий прямое отношение к энергопотреблению вопрос о регулировке частот процессоров.

Обсуждение этого, включая регулировку частот в Linux, вынесено в целый отдельный раздел по четырем причинам:

1. Из-за влияния на энергопотребление.
2. Изменение частот может иметь место у всех процессоров, например и у ARM, а не только у серверных процессоров x86 AMD и Intel.
3. Из-за важности информации об уменьшении повышенных (турбо) тактовых частот при интенсивной вычислительной нагрузке на процессорные ядра (особенно при работе с AVX-512), поскольку это влияет и на возможные оценки пиковой производительности.

   В приведенной Fujitsu информации о своих серверах с процессорами Xeon SPR указывается, что применение режима турбо-частот может приводить и к понижению достигаемой производительности по сравнению с отключением этого режима на вычислительно интенсивных приложениях, например, с рабочими нагрузками на уровне теста HPL [**63**].

   Информация производителя о снижении тактовых частот процессоров часто бывала недоступной, или конфиденциальной. В первом поколении масштабируемых процессоров Xeon использовалось

понятие «базовая частота ядра AVX-512» (об этом и о понижении частоты в турбо-режиме см. [64]). В последних рассматриваемых в обзоре поколениях процессоров часто стали указывать тактовую частоту, поддерживаемую одновременно всеми работающими ядрами.

В Xeon третьего поколения появилось понятие высокоприоритетных и низкоприоритетных ядер, для которых указываются свои разные частоты. Это дает возможность настроить ядра с предоставлением более высоких турбо-частот приоритетным ядрам по сравнению с низкоприоритетным, и относится к технологии Intel Speed Select [65]. Хотя сама эта технология поддерживается и в Xeon 6, указания на высоко- и низкоприоритетные ядра отсутствуют в их спецификациях [66]. Здесь это подробнее не рассматривается, так как относится только к определенным процессорам Xeon, а для использующих распараллеливание задач HPC, где наиболее важно изменение частот при работе с AVX-512, это представляется не так актуальным.

4. В Linux регулировка частоты процессоров была сделана достаточно давно (по сравнению с временами появления новых поколений x86), с ориентацией на универсальный подход, не направленный на конкретные семейства процессоров. Другое дело, что реализации этого с применением также конкретных драйверов ядра Linux от разработчиков соответствующих процессоров давали определенные ограничения (а многие процессоры вообще не позволяли этим пользоваться). Но в последнее время в связи с ростом важности энергопотребления ситуация поменялась в положительную сторону.

Опишем общие применяемые для регулировки частоты разных процессоров подходы и термины, используемые при работе в Linux. Они понадобятся в разделах про конкретные процессоры x86 от AMD и Intel. Описание ориентировано на Linux, понятно, поскольку эта ОС является основной при работе серверов, в частности, с вычислительно сложными приложениями HPC и ИИ.

В ядре Linux для вышеуказанной задачи имеются средства CPUfreq для увеличения или уменьшения частоты процессора в целях повышения производительности или экономии энергии, и соответственно его регуляторы (governors) [67, 68].

В [68] указано о 6 разных регуляторах «целевых направлений» – уровнях абстракции работы процессора, в которых применяются соответствующие алгоритмы, и между которыми можно выбирать при работе в Linux.

Для анализа этого используются C- и P-состояния процессора, которые имеют по нескольку своих уровней. Они давно применяются в исследованиях многоядерных процессоров (см., например, [69]). Эти состояния процессора, как и T-состояния, относятся к интерфейсу `ACPI` (Advanced Configuration and Power Interface) [70].

T-состояния относятся к механизму снижения тактовой частоты, когда температура достигает критически высокой величины. Это кажется естественным для процессоров Intel с реализацией AVX-512. Но Intel с масштабируемыми процессорами Xeon давно предлагает собственные эффективные механизмы управления P-состояниями (см., например, [71]). AMD в описании архитектуры процессоров Zen 4 [23], в которых AVX-512 реализованы в более простом двухстадийном режиме (см. раздел 3.1 далее), о T-состояниях не упоминает, и мы в этом разделе их не рассматриваем.

В конкретных семействах процессоров имеются уточнения/изменения, которые ниже рассматриваются в других разделах. Современная динамически изменяемая информация, в том числе по соответствующим драйверам Linux для разных процессоров, может быть получена из [72]. Вышеуказанные состояния процессоров используются в соответствующих описаниях дистрибутивов Linux, и мы здесь базируемся на информации для широко используемых в области HPC дистрибутивов [73, 74].

C-состояния. Чем больше величина n в состоянии Cn, тем «дальше» процессор от состояния работы и тем меньше потребляемая им энергия.

C0 – процессор включен и работает.

C1 – первое состояние ожидания. Процессор не выполняет команды. Программно останавливает главные внутренние часы [73], но типично не находится в состоянии с пониженным энергопотреблением, и может продолжить работу практически без задержки [74]. В [74] это состояние вкратце называют Halt.

C2 – основные внутренние часы процессора аппаратно остановлены. Но программное состояние сохраняется в норме, и работа может быть возобновлена после прерываний, но с определенной задержкой. В [74] это состояние кратко называют Stop-Clock

C3 – Процессор спит, и все его внутренние часы остановлены. Он не должен соблюдать когерентность кэша [73]. Для пробуждения нужно существенно больше времени, чем из состояния C2. В [74] это состояние кратко называют Sleep.

Состояния C0 и C1 поддерживаются всеми процессорами [73], состояния C2 и C3 относятся к опционным.

P-состояния. Когда процессор работает (находится в состоянии C0), он может находиться в разных состояниях производительности (P-состояниях). Все P-состояния являются рабочими,они связаны с используемой частотой и напряжением процессора. Чем больше величина n в состоянии Pn, тем ниже частота и напряжение,

на которых работает процессор. Хотя P0 всегда относится к состоянию наивысшей производительности (для турбо-режима имеются свои особенности), число и реализация различных P-состояний зависят от конкретных процессоров [73].

Соответственно информация про C- и P-состояния анализируемых в обзоре процессоров будет рассмотрена далее в соответствующих им разделах обзора.

В завершение раздела рассмотрим регуляторы CPUfreq в Linux.

performance – заставляет процессор установить статическую максимально возможную для него частоту. Это ориентировано на случаи, когда процессор практически все время сильно загружен, и никакого сохранения энергии быть не должно.

powersave – частота процессора статически устанавливается в наименьшее возможное значение. В этом случае энергия максимально сохраняется, а производительность понижается. Но, как указано в [73, 74], это не всегда соответствует устремлению к максимальному понижению используемой энергии. Сами соответствующие величины частот в Linux можно изменить [73].

onedemand – этот регулятор нацелен на динамическое изменение частоты в зависимости от текущей нагрузки системы [68]. Понятно, что изменение частоты всегда приводит к определенным задержкам, и при достаточно частом переключении между режимами ожидания и высокой рабочей нагрузки применение этого регулятора неприемлемо. Но он подходит для ситуаций, когда система занята только в определенное время суток. Как указано в [73, 74], при высокой нагрузке на систему регулятор установит максимальную частоту процессора, а при низкой – минимальную.

conservative – этот регулятор близок к ondemand и также предназначается для динамического изменения частот. Но он производит это не скачками от минимальной до максимальной частоты, а более плавно [68]. Этот регулятор подстраивается под тактовую частоту, которую он считает наиболее подходящей для нагрузки, а не просто выбирает между максимумом и минимумом. Хотя это может обеспечить значительную экономию энергопотребления, для этого требуется величины задержки больше, чем у регулятора onedemand [74].

userspace – этот регулятор позволяет процессу или программе устанавливать конфигурируемые в sysfs-файле величины используемых частот [68] и может дать более эффективный баланс производительности и энергопотребления [74].

Есть еще шестой регулятор, *schedutil* [68], но он в [73] и [74] вообще не рассматривается, и здесь анализироваться не будет. Работа регуляторов предполагает не только определенную версию ядра Linux, но и соответствующую реализующую это работу драйверов для конкретных семейств процессоров, что приводит к появлению у них своих особенностей. Поэтому, как уже было сказано выше, далее в обзоре регулировка частот рассматривается для конкретных поколений анализируемых процессоров AMD и Intel.

1.3. Средства разработки программ (SDK, Software Development Kit) для x86

1.3.1. *Традиционные (классические) и новые SDK для x86 от разных разработчиков*

В связи с огромной распространенностью применения x86 за многие годы созданы и используются много разных SDK, которые в той или иной степени применяются для самых разных процессоров x86 разных производителей, и поэтому рассматриваются здесь как часть раздела 2 об общем для процессоров x86.

SDK для x86 – наиболее широко употребляемые и наиболее известные во всем мире, которые регулярно анализируются в публикациях, а их эффективность не только достаточно сильно зависит от ориентации, например, конкретного использующего их создаваемого приложения, но и быстро меняется во времени с появлением новых версий SDK. Их анализ должен был бы включать не только достигаемый уровень оптимизации, но и широту охвата имеющихся возможностей (например, в компиляторах – поддержки новых версий OpenMP или языковых конструкций новых стандартов языков).

Поэтому здесь проводится только краткий анализ с базовой и практически полезной информацией об актуальных для HPC компиляторах, применимых для рассматриваемых в обзоре процессоров, и совсем немного – об основных библиотеках программ. Отладчики и профилировщики в обзоре вообще не рассматриваются. Библиотеки средств распараллеливания и соответственно коммуникаций сообщениями типа MPI вообще почти не упоминаются.

Далее имеются в виду компиляторы только для C/C++ (вообще говоря, и Fortran, но в этом случае отличия относятся главным образом к фронт-энд части компиляторов). Из-за сверхбыстрых изменений SDK практически все цитируемые ссылки здесь ограничены публикациями последних лет.

SDK разных производителей обычно могут применяться как для Xeon, так и для EPYC (про некоторые ограничения этого речь пойдет ниже). SDK от Intel давно относят к лучшим для x86 в мире, и в практическом плане важно отметить доступность этих средств и для применения с EPYC. Но недавно Intel предложила новое оригинальное поколение SDK, oneAPI [**75**]. В настоящее время в мире используется как классический SDK от Intel («классический» – это термин Intel), так и средства oneAPI. Программные средства oneAPI являются огромным новым продуктом Intel, требующим отдельного анализа, и поэтому будут рассмотрены в следующем подразделе. Здесь мы ограничимся традиционными актуальными для HPC и хорошо известными SDK других разработчиков с упоминаниями о новых версиях SDK и разработках. Компиляторы, которые могут использоваться от ПК до суперкомпьютеров, представлены в таблице 3.

Таблица 3. Современные версии актуальных для HPC компиляторов x86

Compiler	GNU[1]	LLVM[2]	Nvida[3]	Cray[4]	AMD[5]	Intel
C/C++	gcc/g++	Clang	nvc/nvc++	craycc	aocc	icc
Fortran	gfortran	Flang	nvfortran	craftn	aocc	ifort

Текущие версии на 3.09.2024: [1] gcc 14; [2] LLVM 19.1.0-rc1; [3] NVHPC 24.7; [4] CPE 24.07; [5] aocc 4.2.

Компилятор gcc 14.2 поддерживает OpenMP 4.5, частично средства OpenMP 5.0[2], часть возможностей OpenMP 5.1 и немного возможностей и от OpenMP 5.2. Компилятор gfortran поддерживает[3] стандарт Fortran 2008 (и соответственно PGAS-модель coarrays) и некоторые возможности Fortran 2018, а также OpenMP 4.5 и некоторые возможности OpenMP 5.1. Начинается и подготовка к поддержке нового стандарта Fortran 2023[4].

Ссылки на различные компоненты SDK от AMD для работы с процессорами различных архитектур Zen общедоступны[5], а общее руководство по SDK для задач HPC с высокопроизводительными процессорами EPYC 9004 представлено в [**76**]. Для также использующей архитектуру Zen 4 серии менее высокопроизводительных процессоров EPYC 8004 (они в обзоре почти не рассматриваются) имеется собственный аналогичный документ по SDK. AMD в своем наборе компиляторов aocc отмечает их ориентацию в первую очередь на процессоры разных поколений Zen и

[2] https://gcc.gnu.org/wiki/openmp, accessed 3.09.2024.
[3] https://gcc.gnu.org/onlinedocs/gfortran/Standards.html, accessed 3.09.2024.
[4] https://gcc.gnu.org/gcc-14/changes.html, accessed 3.09.2024.
[5] https://www.amd.com/en/developer/zen-software-studio.html, accessed 30.10.2024.

задачи HPC [77]. Компиляторы aocc 4.2 базируются на LLVM 16.0.3. Там поддерживается C/C++ 17 и Fortran 2008 (но без средств coarrays). Для C/C++ поддерживается OpenMP 5.0, для Fortran – OpenMP 4.5; подробнее см. в [78].

Для AMD следует указать еще на новые разрабатываемые средства – компилятор на базе LLVM[6] и AOMP[7] – компилятор с открытым исходным кодом на основе Clang/LLVM, ориентированный в первую очередь на применение средств OpenMP на GPU от AMD. Для новейших процессоров EPYC Zen 5 Turin была разработана версия aocc 5.0 на базе LLVM 17.0.6 [79].

По отношению к HPE/Cray ниже приводятся уже данные не о традиционных версиях их компиляторов, которые перестали поставляться, а о новых. В настоящее время HPE предлагает средства SDK, называемые CPE (Cray Programming Environment). Одна из компонент CPE, именуемая CCE (Cray Compiling Environment) [80] включает в себя, в частности, компиляторы, которые поддерживают языки Fortran, C/C++ и UPC (Unified Parallel C). Компилятор Fortran HPE Cray практически полностью поддерживает стандарт Fortran 2018 с некоторыми исключениями. Современные версии HPE Clang C/C++ базируются на Clang/LLVM. Общие программные средства CPE поддерживают также несколько сторонних компиляторов: aocc, компиляторы Intel, GNU и NVIDIA [81].

Здесь не рассматриваются математические библиотеки, средства обмена сообщениями MPI/SHMEM и другие компоненты CPE (см. о них в [82]).

NVHPC 24.7 [83] содержат компиляторы nvc с поддержкой ISO/ANSI C11, nvc++ с поддержкой ISO/ANSI C++17 и nvfortran с поддержкой ISO/ANSI Fortran 2003. Расширения этих компиляторов для поддержки CUDA здесь не рассматриваются.

Из компиляторов Intel здесь мы укажем только на классические компиляторы icc и ifort, которые активно используются во всем мире для трансляции программ, выполняемых на самых разных процессорах x86, не только Intel, но и AMD. Часто предполагается достижение наиболее высокого уровня оптимизации при использовании этих компиляторов Intel, хотя имеется данные и о том, что нередко это не так, особенно при использовании этих компиляторов для расчетов на x86 от AMD.

[6] https://github.com/ROCm/llvm-project, accessed 2.05.2025.
[7] https://github.com/ROCm/aomp, accessed 2.05.2025.

Исследования, сопоставляющие достигаемый уровень оптимизации этими компиляторами для процессоров x86, начали проводиться весьма давно, и по-прежнему активно выполняются и ныне. Приведем несколько примеров. Различные компиляторы и актуальность их применения для AMD Zen 2 (Roma) рассматривались, например, в работе [84]. Также для расчетов на EPYC Roma и Milan исследовалась оптимизация пятью различными компиляторами, в том числе icc и aocc, в работе [85]. Здесь компилятор Intel показал лучшие результаты, опережая aocc, хотя разница в достигаемой производительности обычно была не так велика.

В качестве другого примера укажем недавнюю работу [86], в которой в тестах MD-bench для молекулярной динамики исследовались icc, icx (это новое поколение, из Intel oneAPI), aocc, gcc и clang. Сопоставляются также и разные версии одного и того же компилятора (например, gcc – в [87], но там сравнение проводилось на процессоре Intel Core i7).

Все эти работы включали данные об оптимизации для процессоров Intel и AMD не старше третьего поколения. Быстрый прогресс в аппаратуре рассматриваемых в обзоре процессоров и расширения в них ISA, такие как поддержка AVX-512 в EPYC Zen 4, делают актуальными данные для конкретных рассматриваемых в обзоре поколений x86.

Для EPYC 9004 (модели EPYC 9654) данные об эффективности применения для теста HPL компиляторов aocc, средств библиотек aocl (AMD Optimizing CPU Libraries) и OpenMPI по сравнению с oneAPI Base & HPC Toolkit Classic Compiler 2022.2.0, oneAPI MKL 2022.2.0 (см. о них далее в разделе 1.3.2) и Intel MPI 2021.6 приведены в [88]. В [89] на примере EPYC 9654 сопоставлена получаемая от компиляторов aocc 4.0 и Intel производительность для целого ряда известных приложений, включая молекулярную динамику, CFD и предсказание погоды. Было найдено, что достигаемое различие лежит в пределах нескольких процентов.

В [90] с применением компиляторов aocc 4.0.0 и ifort 2019 сравнивалась достигаемая производительность известного CFD-приложения ANSYS LS-DYNA на двухпроцессорном сервере с EPYC 9654. При компиляции с поддержкой AVX2 чуть более быстродействующий исполняемый код давал aocc, а при использовании AVX-512 чуть быстрее такой код давал компилятор ifort.

Широкий диапазон различных приложений применялся при сравнении оптимизации компиляторами gcc 13.1 и LLVM Clang 16.0 [91]. Хотя разница в среднегеометрической оценке по всем приложениям составила всего несколько процентов, но известное мини-приложение молекулярного

докинга miniBUDE работало в 1.6 раза быстрее при использовании gcc, а тест oneDNN (про библиотеку Intel oneDNN говорится ниже) работал в разы быстрее при компиляции Clang. Приложение молекулярной динамики LAMMPS при использовании обоих компиляторов имело близкую производительность.

Применение среднегеометрической оценки имеет отчасти и ограниченный смысл из-за неясности уровня, который должны бы иметь различные «вклады». В любом случае важными могут оказаться возможные уточнения используемых для конкретных приложений параметров оптимизации. Поэтому по крайней мере для свободно доступных компиляторов целесообразно обеспечивать возможность их выбора программистами.

В связи с расширениями в ISA у процессоров Intel, начиная с Xeon SPR, на AMD Zen 4 не смогут выполняться некоторые использующие эти расширения современные программные средства. И некоторые SDK из состава oneAPI также на Zen 4 не применимы, например, библиотека oneDNN (см. о ней в разделе 1.3.2) для работы с задачами ИИ, которая поддерживает AMX-расширение ISA (см. о нем подробнее в разделе 3.1.2).

Однако AMD предлагает собственную аналогичную библиотеку ZenDNN, оптимизированную для характерных отличий Zen 4 – большого количества ядер и большой емкости кэша L3 [92]. Данные о достигаемой с ZenDNN на Zen 4 производительности имеются, например, в [93].

1.3.2. *Средства oneAPI и набор программных инструментов Intel oneAPI*

Из-за очень большой широты охвата области применения средств oneAPI они обсуждаются здесь в отдельном подразделе. Безусловно, средства oneAPI заслуживают гораздо более подробного рассмотрения, чем будет проведено далее. Однако про их «классическую» часть от Intel немного сказано выше, а большая и очень важная часть компонент oneAPI относятся к задачам ИИ, на которые в обзоре обращается меньше внимания по сравнению с HPC.

С точки зрения автора, наиболее яркие достижения Intel последнего времени лежат как раз в области программного обеспечения, SDK. Здесь необходимо различать реализующийся по исходной инициативе Intel проект oneAPI (проект с открытым исходным кодом с открытым и основанным на стандартах набором интерфейсов, ориентированный на разные типы архитектуры, включая не только процессоры, но и акселераторы), поддерживаемый в рамках группы Unified Acceleration (UXL) Foundation [75], и конкретный оптимизированный свободно доступный программный продукт Intel oneAPI для разных аппаратных средств.

UXL-проект oneAPI во многом ориентируется на стандарт SYCL, расширение C++, которое в настоящее время наиболее актуально для применения в разных акселераторах (например, GPU), хотя может использоваться и на современных процессорах. Intel предложила и реализовала расширение SYCL, DPC++. Некоторую информацию о достигаемой при использовании этих средств производительности в GPU разных разработчиков можно найти в [2]. Кроме DPC++/SYCL, oneAPI включает широкий набор библиотек со стандартизованными интерфейсами, что и породило API в названии. Здесь, естественно, рассматриваются только относящиеся к x86 средства Intel oneAPI. Везде далее под oneAPI подразумевается именно Intel oneAPI для x86.

Безусловно, в oneAPI представлены аналоги знаменитых классических программных средств Intel. Например, библиотека MKL теперь называется oneMKL. Она, как известно, включает кроме программ для векторной математики, плотной и разреженной линейной алгебры также средства для преобразования Фурье и генераторы случайных чисел. Классические компиляторы C++ и Fortran также входят в состав oneAPI, но развиваться далее будут компиляторы oneAPI на основе LLVM. Это не означает, что все новые компиляторы oneAPI всегда превосходят по уровню оптимизации классические компиляторы Intel (см. [94]), но это может зависеть от версии.

Intel активно проводит ориентацию своих процессоров четвертого и пятого поколений на решение задач ИИ, соответственно в oneAPI представлены актуальные для этой области библиотеки. Упомянем, например, библиотеку oneAPI Data Analytics Library (oneDAL) для машинного обучения, ускоряющую анализ больших данных на всех стадиях: получение данных из источника данных, предварительную обработку, преобразование, интеллектуальный анализ данных, моделирование, проверку и принятие решений. OneAPI Deep Neural Network Library (oneDNN) предоставляет основные строительные блоки, используемые в приложениях глубокого обучения.

Мы приведем здесь только ссылки на документацию по средствам oneAPI для HPC [95] и для ИИ [96]. Хотя направления работы Intel все больше устремляются к задачам ИИ, можно отметить членство Intel в сообществе OpenHPC, ориентированном на объединение общих программных компонентов с открытым исходным текстом, нужных для развертывания и управления Linux-кластерами [97].

Текущие (доступные на 5.09.2024) документации для новых компиляторов Intel относятся к версиям 2024.2.1 для DPC++/C++ и 2024.2.0 для Fortran (ifx), а для классического компилятора – к версии 2021.1.0 для icc и 2024.2 – для ifort. Полные описания oneAPI доступны в [98]. Что касается ifx, то имеются данные о росте производительности при его использовании в 2P-сервере с Xeon Max 9480 на величину до 17% по сравнению с классическим ifort (в основанных на Fortran тестах из состава SpecOMP 2012) [99], хотя в некоторых тестах бывало понижение на 3%.

Говоря о применении SDK от Intel для работы с EPYC Zen 4, следует отметить, что самые высокие показатели производительности процессоров AMD Genoa (Zen 4, содержащих наиболее высокопроизводительные ядра – см. далее в разделе 2.2) в данных тестов SPECcpu 2017 [51] получены с использованием средств aocc. Однако в суперкомпьютерных центрах, в том числе содержащих EPYC Genoa в узлах, возможно чаще используют компиляторы Intel. Оптимальным и часто применяемым в суперкомпьютерных центрах является предложение целого набора разных компиляторов.

Что касается математических библиотек, то кроме применения открыто доступных библиотек с Zen 4 может использоваться и MKL. Библиотека MKL для 2P-сервера с EPYC 9334 в [100] для HPL давала на 5% более высокую производительность, чем свободно доступная библиотека OpenBLAS. Известное отключение проверки идентификатора процессора в MKL на этом сервере увеличило производительности на 20%.

В отчете [88] на 2P-сервере с EPYC 9654 была получена при использовании aocc, aocl и OpenMPI отмечена более высокая производительность в HPL, чем при работе с oneAPI Base & HPC Toolkit Classic Compiler 2022.2.0, oneAPI MKL 2022.2.0, и Intel MPI 2021.6. Хотя производительность библиотек здесь могла оказаться определяющей, такой результат может быть связан с отсутствием отключения проверки идентификатора для Zen 4.

2. Процессоры AMD EPYC четвертого поколения (архитектуры Zen 4)

В Zen 4, естественно, крайне много общего с архитектурами предыдущих поколений процессоров AMD, особенно с Zen 3. Понятно, что много ярких особенностей Zen 4 обусловлены давним применением AMD многокристальной технологии. Но здесь укажем только одну общую

особенность, давно имеющуюся в серверных процессорах x86 как от AMD, так и от Intel – поддержку SMT (конкретно – двух логических ядер на одном физическом ядре). Про улучшения за счет применения SMT в Zen 4 по сравнению с Zen 3 см. в [22]. Далее об SMT речь в основном пойдет только в разделе 2.4 при рассмотрении влияния SMT на производительность. Для многих задач HPC включение SMT неэффективно.

2.1. Микроархитектура и ISA ядер Zen 4

Рассмотрение микроархитектуры процессоров мы начнем с микроархитектуры ядер. Такой анализ важен в более широком плане, чем тематика данного обзора, поскольку эти ядра применяются также и в других (не серверных) процессорах AMD. Достижение высокой производительности каждого ядра важно не просто из-за стремления к высокой производительности всего процессора, но особенно для ситуаций, когда за лицензию на программное обеспечение платится по числу используемых ядер. Информация в этом разделе будет также включать данные об усовершенствованиях в ISA, поскольку команды выполняются именно ядрами процессоров.

Основной публикацией AMD по микроархитектуре ядер Zen 4 можно считать работу [101]. Каждое ядро включает в себя 1 МБ частного кэша L2, что вдвое больше, чем в Zen 3. Улучшение IPC по сравнению с предыдущим поколением в среднем для однопоточных настольных приложений составляет 13%. Ядро Zen 4 может работать на частоте 5.7 ГГц. Относительно Zen 3 повышена как однопоточная производительность (более чем на 29%), так и производительность на ватт в многопоточных рабочих нагрузках [101].

Построение фронт-энд части Zen 4 (см. блок-схему на рисунке 4) по сути кардинально не отличается по сравнению с имеющейся для Zen 2 блок-схемой на известном сайте wikichip [102], где размещается детальная информация о микроархитектуре процессоров. Однако в [102] для Zen 2 рисунок более детален и охватывает большую часть ядра. Аналогичная информация для Zen 3 и Zen 4 в [103, 104] рассматриваются менее детально.

В Zen 4 за такт может диспетчироваться до шести целочисленных операций, и выполняться до трех загрузок и до двух сохранений. Точность предсказания переходов улучшена по сравнению с Zen 3. Увеличены также размеры буферов по всему ядру. Увеличена емкость кэша предварительно декодированных инструкций (Op Cache на рисунке 4), емкости очереди завершения (retire) обработки микроопераций, и файлов целочисленных регистров и регистров с плавающей запятой [101].

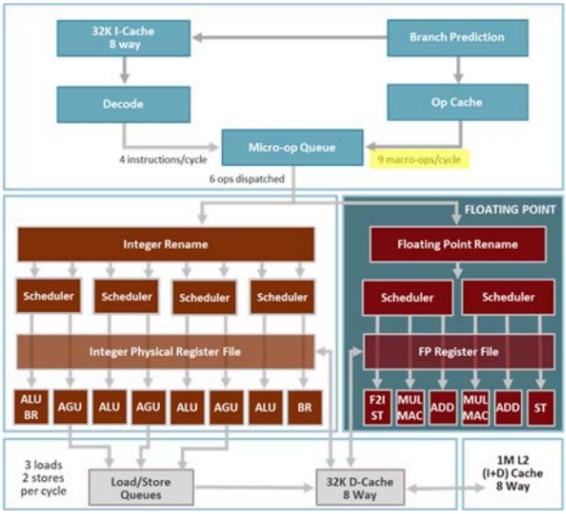


Рисунок 4. Микроархитектура ядер Zen 4 (рисунок из [101])

Что касается фронт-энд части, то в [22] относительно устройства выборки команд отмечены увеличение пропускных способностей очереди выборки команд и их диспетчирования, а также емкости кэша микроопераций. Ряд из этих данных отображены в таблице 4. На отнесенном к исполнительным устройствам слайде в [22] отмечено увеличение очередей загрузки/сохранения, обоих уровней буферов быстрой перезагрузки страниц (L1 и L2 DTLB) и др. (см. также таблицу 4).

Собственно про блоки выполнения мы скажем чуть ниже отдельно, поскольку в Zen 4 была расширена ISA и появилась частичная поддержка AVX-512. Интерес к более подробным, чем ставшие свободно доступными данным о микроархитектуре Zen 4, включая детальные блок-схемы, привел к появлению работ по установлению соответствующей информации с помощью применения микротестов (см. [106]). Подробные количественные данные о микроархитектуре Zen 4 имеются на сайте wikichip [104]; они были частично использованы и при построении таблицы 4.

Естественно, на увеличение производительности ядер Zen 4 относительно ядер Zen 3 влияет и увеличенная емкость кэша L2 (см. таблицу 4), и доли общего для ядер кэша L3, приходящегося на одно ядро (это будет рассмотрено далее).

В интегральном плане отмечается, что указанные выше и приведенные в таблице 4 усовершенствования микроархитектуры Zen 4 относительно

Таблица 4. Сравнение показателей микроархитектуры ядер Zen 3 и Zen 4

	Zen 3	Zen 4
Очередь загрузки, записей	72	88
Очередь сохранения, записей	64	64
Размер кэша микроопераций	4096	6912
8-канальный наборно-ассоциативный кэш L1 I/D	32/32 КБ	32/32 КБ
8-канальный наборно-ассоциативный кэш L2	512 КБ	1 МБ
Кэш L3 на ядро, МБ[1]	4	4
L1 DTLB[2], записей	64	72
L2 DTLB[3], записей	2048	3072
Задержка кэша L2, тактов	от 12	от 14
Задержка кэша L3[4], тактов	46	50
Ширина выдачи команд (INT+FP/SIMD)	10+6	10+6
Файл целочисленных регистров, записей	192	224
Целочисленный планировщик, записей	96	96
Файл регистров FP, записей	160	192
Буфер переупорядочивания, записей	256	320
Задержка FADD/FMUL/FMA[5], тактов	3/3/4	3/3/4
Записей в L1 BTB	2x1k	2x1.5k
Записей в L2 BTB	2x6.5k	2x7k1

[1] 16-канальный наборно-ассоциативный;
[2] полностью ассоциативный;
[3] Zen 4: 24-канальный наборно-ассоциативный (в Zen 3: 16-канальный наборнно-ассоциативный);
[4] от загрузки до начала использования (в среднем). BTB – branch target buffer.
[5] векторных команд с плавающей запятой сложить/умножить/умножить-и-сложить

Источники данных: [103, 104], рисунок-таблица на сайте базы данных процессоров [105].

Zen 3 дают увеличение IPC на 14% [22, 23, 107]. Данные о том, какие усовершенствования микроархитектуры ядер дают какой вклад в рост IPC, были представлены на слайде доклада AMD, который носил сначала конфиденциальный характер, но потом стал свободно доступен на ряде сайтов (см., например, [108]). Но эти данные со среднегеометрической оценкой по 22 различным рабочим нагрузкам относятся к ядрам Zen 4 в процессорах AMD Ryzen, и очевидно, что использовались и соответствующие характерные для ПК рабочие нагрузки.

По этим данным, наименьший вклад в рост IPC вносит емкость кэша L2, почти столько же – блок выполнения, раза в два больше дали загрузка/сохранение, почти столько же – предсказание переходов, и еще раза в два больше дали другие блоки фронт-энд части ядер.

К этим данным об IPC можно также добавить полученную в [88] близкую оценку роста IPC (на 12%) в EPYC 9654 относительно Zen 3 Milan-X (EPYC 7773X) в известном квантовохимическом приложении Gaussian 16.

Исполнительные устройства в Zen 4 были существенно модернизированы относительно Zen 3 благодаря внедрению поддержки расширения ISA, AVX-512, которое давно используется в процессорах Xeon и давало им ранее четкие преимущества по пиковой производительности. Этот факт одновременно сопровождался до самого последнего времени другими фактами – отсутствиями аппаратной поддержки 512-битных векторов процессорами как AMD – в EPYC, так и серверными ARM-процессорами, включая и Nvidia Grace с архитектурой Neoverse V2 (A64FX являются в этом плане исключением [7]).

К этому есть понятные причины – возникающая из-за поддержки AVX-512 необходимость дополнительной площади кристалла, и соответственно увеличение стоимости и энергопотребления. Разработчики, естественно, сравнивали возможное удорожание процессора и с рост производительности для широко используемых приложений, в том числе и в HPC-области. Хотя для приложений, производительность которых при этом сильно возрастает, работа с AVX-512 может давать и повышение энергоэффективности.

Самым очевидным, с точки зрения автора, где очень полезно иметь 512-битовые вектора, в HPC-области является умножение больших плотных матриц (DGEMM), что актуально, например, для явно учитывающих корреляцию методах квантовой химии (но не для широко распространенных квантовохимических методов DFT). В качестве самого знаменитого теста производительности, где важно умножение матриц, обычно упоминается HPL [109].

В Zen 4 AMD реализовала аппаратную поддержку части (с точки зрения автора – наиболее важную) «подмножеств» из полного расширения Intel AVX-512 (про эти подмножества см., например, [110]).

Кроме естественной для HPC реализации в AVX-512 операций умножения-и-сложения векторов (fused multiply-add, `FMA`) в формате FP64, в Zen 4 поддерживается работа с актуальным для ИИ форматом низкой точности BF16 (который часто считают более полезным, чем FP16). Другая важнейшая для задач ИИ реализованная в Zen 4 поддержка AVX-512 – выполнение векторизованных операций нейронной сети VNNI [22, 23]. Остальные поддерживаемые в Zen 4 возможности AVX-512, в том числе и для криптографии, здесь вообще не рассматриваются.

Для обсуждения способа аппаратной реализации AVX-512 в Zen 4 нужно обратить внимание на то, как это было реализовано ранее в Xeon.

С точки зрения сопоставления возможной производительности при работе с AVX-512 в Xeon, давно поддерживающих реализацию AVX-512, надо иметь в виду, что ее применение с операциями FMA для задач HPC приводило там к сильному понижению используемой тактовой частоты (по сравнению с указанной Intel в спецификации турбо-частотой) сразу при выполнении таких команд, а не после повышения выделяемой процессором энергии.

Отсутствие детального описания Intel механизмов понижения частоты при работе с AVX-512 приводило к появлению исследований этого (не обязательно в виде научных публикаций) и работ, нацеленных на более эффективное применение AVX-512, несмотря на понижение частоты (см., например, [**111**–**113**]). Естественно, стала изучаться и достигаемая при этом энергоэффективность [**114**].

Однако относительно «десктопных» процессоров Intel ICL имеется информация о кардинальном улучшении механизма уменьшения частоты [**115**]; в Xeon ICL механизм понижения частоты также был улучшен (см., например, [**116**]). Но на конец 2023 года детальные данные Intel по поведению турбо-частот при работе с AVX-512 относились к разряду конфиденциальных [**117**].

Целесообразность аппаратной поддержки процессорами матричных операций, особенно для HPC-области, вообще подвергается сомнению [**109**]. Встает вопрос о большей эффективности применения GPU вместо процессоров для сложных расчетов в приложениях, где производительность сильно растет при работе с 512-битными векторами. Однако AMD с ориентацией на задачи HPC (и ИИ) в своих новых процессорах Zen 4 обеспечила поддержку AVX-512.

Аппаратная реализация AMD расширения AVX-512 существенно отличается от ее реализации в Xeon Platinum и большинстве моделей Xeon Gold. AMD пошла по пути экономичного решения, при котором может не требоваться сильное понижение тактовых частот при работе с AVX-512, как у Xeon. Однако при этом пиковая производительность ядер (из-за более низкого числа FLOPS/такт в Zen 4) оказывается все-таки ниже, чем у Intel.

Основная информация от AMD о том, как была аппаратно реализована поддержка AVX-512 в Zen 4, представлена в [**22**, **23**]. Хотя в Zen 4 имеется необходимый для AVX-512 набор 512-битных регистров, но

далее используются 256-битные каналы и 256-битные блоки выполнения. Выполнение 512-битных операций реализовано в виде двух последовательных 256-битных операций, выполняемых в двух соседних тактах. С точки зрения разработчиков, такая реализация способствует высокой энергоэффективности [**22**].

Появляются и предположения, что EPYC 9004 могут при этом проводить больше времени в турбо-режиме [**100**]. И, естественно, получается возможным выполнять на Zen 4 двоичный код приложений, использующих AVX-512.

Но AMD еще усилила свои исполнительные устройства для AVX-512. В ядрах Xeon Platinum и большинстве Xeon Gold имеется по два устройства, способных выполнять 512-битные `FMA`-операции, что для FP64 дает 32 FLOPS/такт. Это выглядит естественно для актуальных для HPC DGEMM-умножений матриц, способных достигать производительности, максимально приближенной к пиковому значению.

Каждое ядро Zen 4 также как у Intel имеет по два исполнительных устройства, но их аппаратура содержит дополнительный к `FMA` векторный блок сложения [**22**]. Соответственно, каждое физически 256-битное исполнительное устройство выдает три результата над векторами длиной четыре числа FP64 за такт, а два исполнительных устройства дают 24 FLOPS/такт.

Для реализации DGEMM это не выглядит естественным, и не способствует достижению в DGEMM производительности, близкой к формальной пиковой величине. Можно выдвигать предположения, как такие расширенные возможности Zen 4 могут эффективно использоваться. Согласно [**100**], в однопроцессорной (`1P`) конфигурации в тесте HPL это дало возможность получить производительность выше пиковой величины, рассчитанной без учета наличия дополнительного блока сложения (т.е. с 16 FLOPS/такт).

В разделе 2.4.3 для двухпроцессорной (`2P`) конфигурации представлены подтверждающие такой уровень достигаемой производительности данные от AMD. Для уточнения ситуации нужны дополнительные исследования. Данные о производительности EPYC 9004 в DGEMM и HPL рассматриваются далее в разделе 2.4.3.

Отставание ядер Zen 4 от ядер масштабируемых процессоров Xeon четвертого и пятого поколений в числе FLOPS/такт может компенсироваться большими тактовыми частотами, большим числом ядер или другими преимуществами Zen 4 – но это относится к сравнительным данным о реально достигаемой производительности, которая будет обсуждаться далее.

В Xeon SPR Intel уже реализовала новое расширение ISA, AMX (Advanced Matrix Extensions) [118] с поддержкой матричных операций пониженной точности для задач ИИ. Это является важным преимуществом для этих процессоров Xeon в отношении задач ИИ. Что касается HPC, то AMX может потенциально стать актуальным в сильно ограниченном масштабе – в случае возможности эффективного использования для эмуляции расчетов более высокой точности. Работы в таком направлении сейчас ведутся, но в основном с ориентацией на тензорные ядра в GPU – см. об этом, например, в [2, 109].

В заключение этого раздела необходимо отметить, что у AMD есть еще чуть отличные от Zen 4 процессорные ядра Zen 4c. Содержащий до восьми ядер Zen 4c комплекс ядер (Core Complex, CCX) имеет расположенный в нем общий кэш L3 емкостью 16 МБ – в два раза меньше, чем у ядер Zen 4 [23]. Комплексы ядер CCX будут обсуждаться в следующем разделе 2.2.

В отличие от оптимизированных для высокой производительности ядер Zen 4, ядра Zen 4c оптимизированы по площади кристалла для энергоэффективности [23, 119]. Ядра Zen 4c используются не во всех типах процессоров EPYC Zen 4 (см. таблицу 5 ниже), и к обсуждению здесь микроархитектуры ядер они ничего важного, кроме вышесказанного, не добавляют. Подробнее о ядрах Zen 4c см. [119].

2.2. Микроархитектура процессоров EPYC Zen 4

Основным источником данных для всего дальнейшего анализа в этом разделе является [23]. В случае, если информация была получена из других источников, на них дается соответствующая ссылка.

Величина IPC, возрастающая благодаря проводимым усовершенствованиям в ядрах – это только один интегральный (для ядра) показатель среди информации о многих компонентах процессора, на которые в первую очередь обращают внимание производители, в том числе приводя табличные данные о достигнутом прогрессе в новых поколениях x86. Пожалуй, больше внимания, особенно AMD, в последнее время обращается на рост числа процессорных ядер.

Иерархия построения процессоров EPYC из отдельных ядер включает несколько уровней, что связано с большим количеством используемых ядер и применением многокристальной технологии (чиплетов).

Следующий уровень иерархии над ядрами называется комплексом ядер, CCX. Из них строятся кристаллы CCD (Core Complex Die), а несколько кристаллов CCD образуют целый процессор. Прежде чем приступить

Таблица 5. Спецификации процессоров Zen 4

	Серия EPYC 8004 (1 сокет)	Серия EPYC 9004 (2 сокета – кроме Genoa 9004P)		
Кодовое имя	Siena	Bergamo	Genoa	Genoa 9004F/Genoa 9004X
Технология	TSMC 5 нм (6 нм для IOD)			
Сокет	SP6	SP5	SP5	SP5
Процессорные ядра	Zen 4c	Zen 4c	Zen 4	Zen 4
Нумерация моделей	8004P, 8004PN	9704	9104-9604[5]	9004F/9004X
Максимальное число ядер (нитей)	64(128)	128 (256)	96 (192)	48(96)/96(192)
Максимальное число ядер на CCX	8	8	8	8
Максимальное число CCD	4	8	12	8/12
Максимальное число CCX на CCD	2	2	1	1
Максимальная емкость кэша L3 (на CCX), МБ	128(16)	256 (16)	384(32)	256(32)/1152(96)
Максимальное число каналов DDR5	6	12	12	12
Максимальная пропускная способность памяти, ГБ/с[1]	230.4	460.8	460.8	460.8
Максимальная емкость памяти на сокет, ТБ	3	6	6	6
Линий PCIe-v5 (максимум)	96	128	128	128
Линий CXL[2]1.1+ (максимум)	48	64	64	64
Максимальная часота, ГГц[3]	3.15	3.15	4.15	4.4/4.2
Максимальная boost-частота всех ядер, ГГц	3.1	3.1	3.55[6]	3.95[7]/3.7[8]
Пиковая производительность, GFLOPS[4]	3532.8	6912	5529.6	4147.2/5875.2

Данные из [22, 49, 120–122].

[1] Кроме используемой DDR5-4800, по данным [123] ожидается и DDR5-5200;

[2] Compute eXpress Links;

[3] Здесь приводится boost-величины частот;

[4] Рассчитана на основании базовой частоты для старших моделей – EPYC 8534P, 9754, 9654 и 9474F/9684X соответственно;

[5] Использована традиционная для нумерации моделей замена разных цифр на 0;

[6] for EPYC 9654, 3.9 for EPYC 9254;

[7] for 9474F, 4.15 for EPYC 9174F;

[8] for EPYC 9684X, 4.20 for EPYC 9184X.

Примечание. IOD (I/O Die) – кристалл ввода-вывода, см. об этом ниже. Приводимые в таблице характеризуемые как максимальные значения могут не достигаться одновременно (в одной конкретной модели EPYC).

к анализу этой иерархии, укажем на общую для этого обсуждения и последующих анализов производительности таблицу 5.

Особенности процессоров серии EPYC 4004 описаны далее в отдельном подразделе. EPYC 4004 обладают практически всеми рассматриваемыми далее особенностями микроархитектуры, отличаясь в основном чисто количественными показателями.

К Zen 4 относятся три серии серверных процессоров AMD – 9004, 8004 (см. рисунки 5–7) и 4004.

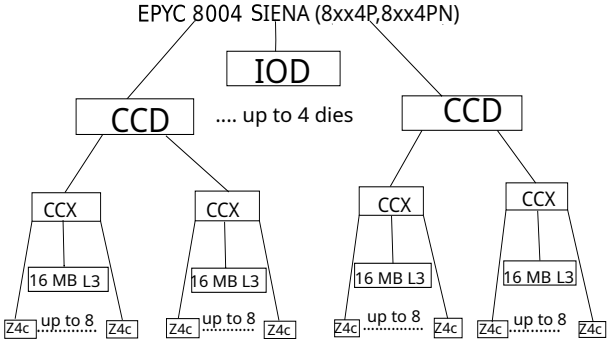


Рисунок 5. Иерархия построения процессоров Siena из ядер (Z4c на рисунке). Кэш L3 – общий для всего процессора

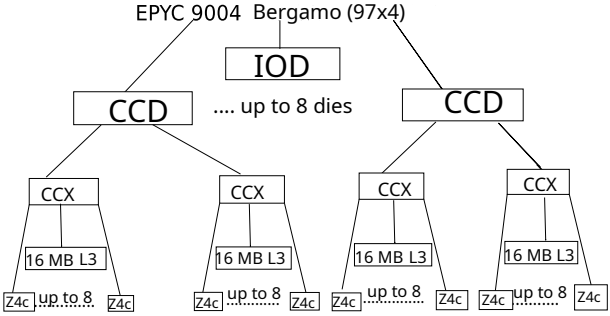


Рисунок 6. Иерархия построения процессоров Bergamo из ядер (Z4c на рисунке). Кэш L3 – общий для всего процессора

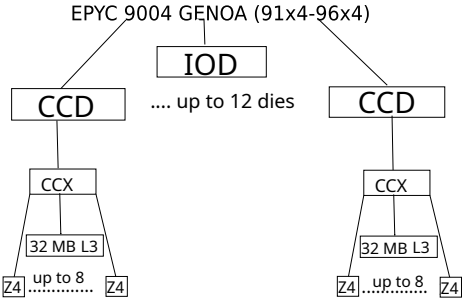


Рисунок 7. Иерархия построения процессоров Genoa из ядер (Z4 на рисунке). Кэш L3 – общий для всего процессора

Процессоры из серии 9004 используют большой сокет SP5 (LGA 6096 [**124**]), число контактов для микросхемы в котором существенно

больше, чем в пятым поколении масштабируемых процессоров Xeon. В серии 9004 имеются модели, которые могут работать только в 1P-конфигурации или в 1P- и 2P-вариантах. В процессорах серии 9004 могут использоваться ядра Zen 4 или ядра Zen 4c.

Процессоры серии 8004 (с кодовым словом Siena) с более низким TDP и более простым сокетом SP6 используют ядра Zen 4c, содержащие в два раза меньше емкость кэша L3, чем в Zen 4. Siena могут работать только в 1P-серверах. Серия 8004 ориентирована на системы с небольшими физическими размерами и работающими в экологически сложных условиях [**23**]. Модели серии 8004 бывают двух классов – с номерами 8004P, и с номерами 8004PN. В последнем классе моделей поддерживаются типичные для телекоммуникационных компаний США условия NEBS (Network Equipment-Building System) [**125**].

Процессоры серии 4004 (с кодовым словом Raphael) относятся к классу серверов с наименьшей (по сравнению с другими сериями) производительностью и определенными отличиями в микроархитектуре от двух других серий. EPYC 4004 не предполагаются для применения в областях HPC и ИИ, на которые в первую очередь ориентируется обзор, и их особенности рассматриваются далее вкратце в отдельном подразделе.

В процессорах Siena до восьми ядер Zen 4c с общим кэшем L3 емкостью 16 МБ объединяются в виде CCX, а до двух CCX формируют кристалл CCD с кэшем L3 емкостью 32 МБ. Наконец, до четырех кристаллов CCD вместе с кристаллом ввода-вывода IOD образуют процессор (см. рисунок 5). До восьми ядер на CCX дают до 16 ядер на CCD и до 64 ядер на процессор.

Общая емкость кэша L3 соответственно составляет до 128 МБ; он разделяется всеми ядрами Zen 4c. Другие данные по процессорам Siena приведены в таблице 5. Далее в обзоре эти процессоры практически не рассматриваются, поскольку они не соответствуют основной его тематике.

Процессоры серии 9004 с кодовым словом Bergamo также строятся на ядрах Zen 4c (см. рисунок 6).

В Bergamo используется такая же иерархия вплоть до кристалла CCD, но процессор может содержать не до четырех, а до восьми кристаллов CCD и соответственно до 128 ядер Zen 4c с общей емкостью кэша L3 до 256 МБ. Другие данные, в том числе о пиковой производительности для FP64 и используемой памяти, приведены в таблице 5.

Процессоры семейства Bergamo нацелены на масштабирование с повышенной энергоэффективностью, и их естественным применением являются задачи облачных технологий [**22**, **107**]. Процессоры серии 9004 с кодовым словом Genoa строятся на ядрах Zen 4 (см. рисунок 7). В Genoa до восьми ядер Zen 4 образуют CCX с общей емкостью кэша L3 до 32 МБ, в два раза

большей, чем в CCX на базе Zen 4c. Кристалл CCD содержит поэтому не два, а один CCX. Весь процессор может содержать до 12 кристаллов CCD и соответственно до 96 ядер Zen 4 с общей емкостью кэша L3 до 384 МБ.

Кроме того, в серии 9004 имеются модели процессоров с кодовым словом Genoa-X. В них используется технология AMD 3D V-cache, которая ранее применялась и в EPYC Zen 3 (Milan). При использовании этой технологии поверх каждого кристалла CCD размещается дополнительный кэш, так что общая емкость кэша L3 на CCD в Genoa-X составляет 96 МБ. Соответственно на процессоре с 12 CCD суммарная емкость кэша L3 (он является общим для всех ядер) составляет 1152 МБ [**23**].

В заключение этого общего введения в иерархическое построение микроархитектуры различных серий процессоров Zen 4 рассмотрим таблицу 6, содержащую конкретные модели этих процессоров, интересные для нашего обзора. Но прежде проясним систему нумерации всех процессоров Zen 4.

Номер модели процессора состоит из четырех десятичных цифр, последняя из которых, равная 4, и означает отнесение модели к архитектуре Zen 4. Про смысл в первой цифре, 8 или 9 для серий 8004 и 9004, только что говорилось выше.

Сконцентрируемся на моделях серии 9004, номер каждой из которых представляется в виде $9KL4$, где K и L – десятичные цифры. С увеличением K от 0 до 6 растет число ядер (C), доступных в процессоре: $C = 8$ при $K = 0$, $C = 16$ при $K = 1$, C = 24 при $K = 2$, $C = 32$ при $K = 3$, C = 48 при $K = 4$, $C = 64$ при $K = 5$ и C бывает от 84 до 96 при $K = 6$.

Увеличение цифры L от 1 до восьми в номере модели соответствует росту производительности [**121**]. Кроме того, в конце номера может быть дополнительная буква F для моделей, отличающихся повышенной тактовой частотой, или буква X для Genoa-X с использованием 3D V-cache, или буква P для моделей, которые могут работать только в 1P- серверах. Для серии 8004 используется аналогичная система нумерации.

Причиной формирования такого большого количества разных моделей процессоров является их ориентация на разные области возможного применения и стоимостные показатели. Это стало возможным вследствие модульного подхода AMD с использованием многокристальной технологии и иерархического построения процессоров, описанного выше.

Intel в рассматриваемых далее процессорах Xeon также предлагает огромное количество самых разных моделей для разных применений. Однако для понимания SKU вышеописанный модульный подход AMD представляется более четким и лучше воспринимаемым.

Таблица 6. Модели серверных процессоров EPYC Zen 4 Номер

Номер модели	Кодовое имя	Число ядер	Частота, ГГц (Base/Boost)	Boost-частота всех ядер	Емкость кэша L3, МБ	TDP, Вт[1]	Цена[2]	Пиковая производительность (FP64, GFLOPS)[3]
9754	Bergamo	128	2.25/3.1	3.1	256	360	$11900	6912
9684X	Genoa-X	96	2.55/3.7	3.42	1152	400	$14756	5875.2
9654	Genoa	96	2.4/3.7	3.55	384	360	$11805	5529.6
9554	Genoa	64	3.1/3.75	—	256	360	$9087	4761.6
9534	Genoa	64	2.45/3.7	3.55	256	280	$8803	3763.2
9474F	Genoa	48	3.6/4.1	3.95	256	360	$6780	4147.2
9454	Genoa	48	2.75/3.8	3.65	256	290	$5225	3168
9384X	Genoa-X	32	3.1/3.9	3.5	768	320	$5529	2380.8
9374F	Genoa	32	3.85/4.3	4.1	256	320	$4850	2956.8
9354	Genoa	32	3.25/3.8	3.75	256	280	$3420	2496
9334	Genoa	32	2.7/3.9	3.85	128	210	$2990	2073.6
9274F	Genoa	24	4.05/4.3	4.1	256	320	$3060	2332.8
9174F	Genoa	16	4.1/4.4	4.15	256	320	$3850	1574.4
8024P	Siena	8	2.4/3.0	2.95	32	90	$409	460.8

Данные из [**122**] и [**125**] от 9.04.2024.

[1] значение по умолчанию; настраиваемые значения см. в [**125**, **126**];

[2] данные из [**49**] от 23.04.2024;

[3] рассчитаны автором для базовой частоты.

В таблице представлены модели, производительность которых будет обсуждаться далее, или которые мы считаем интересными для данного обзора.

Кристалл IOD и Infinity Fabric. Все исполнительные блоки, определяющие производительность процессора, по крайней мере относительно пиковых величин, расположены в ядрах и уже рассмотрены выше. Что касается иерархии памяти, то вплоть до уровня кэша L2 все также относится к ядрам.

Блоки общего для всего процессора кэша L3 также расположены в процессорных ядрах, как и собственно оперативная память также общая для всего EPYC. Для доступа ядер к памяти используется и межсоединение Infinity Fabric (далее в этом разделе и разделе 3 сокращенно именуется IF), которое связывает между собой CCD. Оно соединяет друг с другом все кристаллы (чиплеты) [**121**]. IF, как и контроллеры памяти, в Zen 4 интегрировано в блоке IOD. Хотя IOD реализует очень большое количество разных функций (что позволяет говорить о Zen 4 как о SoC), он несколько проще других кристаллов в Zen 4, и сконструирован с применением более дешевой технологии 6 нм. Блоки IOD и CCD как разные кристаллы могут изменяться и усовершенствоваться независимо.

Общее представление о компонентах IOD дает рисунок 8, а общую иллюстрацию того, как IOD задействован для соответствующих функций в разных сериях процессоров, дает рисунок 9.

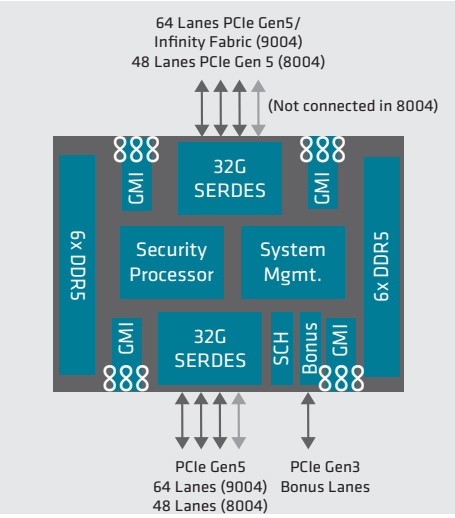


Рисунок 8. IOD (рисунок из [**23**])

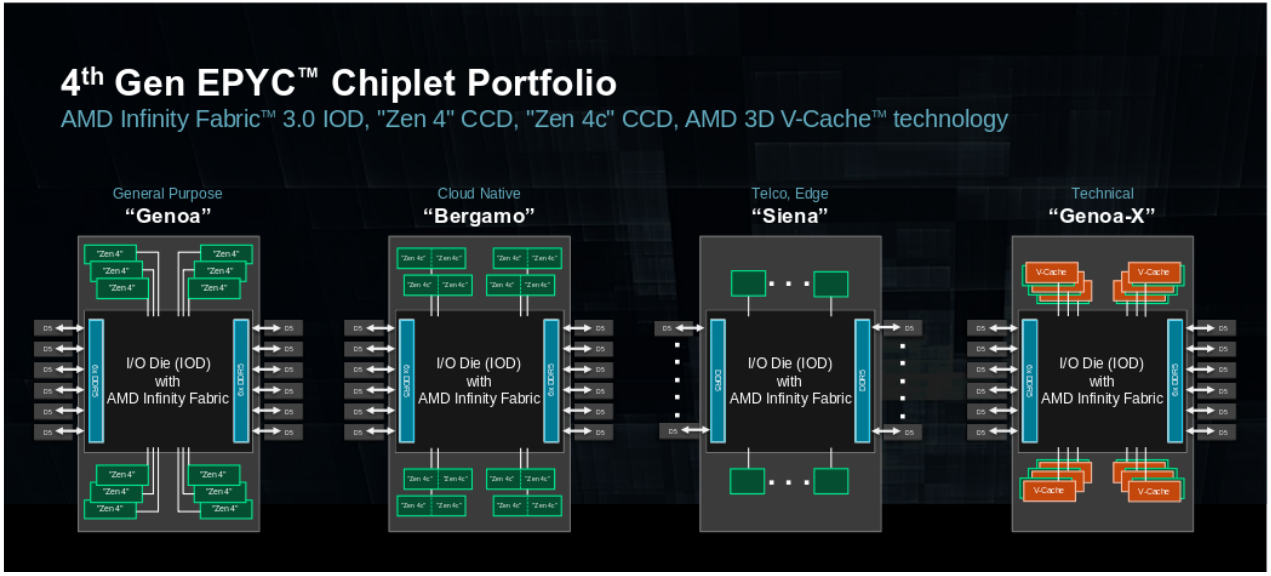


Рисунок 9. Использование Infinity Fabric и `IOD` в разных сериях процессоров EPYC Zen 4 (рисунок из [22])

В майской версии 2024 года документа [23] (в обзоре применялась в основном сентябрьская версия 2023 года) появилась также информация об еще одной компоненте в IOD, блоке управления системой – SMU (System Management Unit), который контролирует распределение мощности на IOD, CCD и ядра, поддерживая процессор в пределах его параметров, в том числе заданных с помощью настроек управления питанием. Об этом речь пойдет ниже при обсуждении TDP.

Технология IF используется AMD не только во всех поколениях процессоров Zen, но и в своих GPU [2] и постоянно усовершенствуется. Растет ее пропускная способность. В Zen 4 применяется версия IF 3.0, в которой используются линии, работающие на скоростях на 78% быстрее, чем в IF 2.0, применявшейся в Zen 3 [127].

Топология IF конфигурируема. Базовая часть IF, Infinity Scalable Data Fabric (SDF) [128], нацелена на обеспечение масштабирования межсоединения. Она основана на более ранней технологии Hyper Transport, которая исходно ориентировалась на высокую пропускную способность и низкую задержку. SDF передает данные между конечными точками, например, между узлами NUMA. SDF может иметь точки подключения для PCIe PHY, контроллеров памяти, хаба USB, а также применяется для обеспечения когерентности кэша. SDF связывается с различными сериализаторами/десериализаторами (SerDes). Блок SCH на рисунке 8 отвечает за поддержку интерфейсов разных шин – I2C, SMBus, SPI/eSPI и т. д. Один из блоков IF, в частности, управляет температурой и электропитанием [128].

Поддержание на физическом уровне PCIe PHY дает IF очевидные преимущества в гибкости. Часть поддерживаемого в IOD большого набора линий SerDes используется для работы с SATA, часть – для работы с CXL 1.1+, часть – для работы с PCIe, часть – для работы с IF. Поскольку технология IF применяется и для межсоединения процессоров в серверах, имеется возможность гибкого конфигурирования для конкретных серверов – часть линий PCIe на физическом уровне переключаемы на работу с каналами IF [23].

Конкретная конфигурация и применение IOD зависят и от используемого разъема процессора EPYC, и от конкретной модели процессора, и от конфигурирования IOD в конкретной модели сервера. Эта гибкость позволяет в конечном счете эффективно создавать конкретную модель сервера для определенной области его применения. Все нижеследующие уточнения базируются на данных [23], а для расширений этой информации приводятся соответствующие дополнительные ссылки.

В EPYC 9004 по сравнению с Zen 3 удвоена пропускная способность ввода-вывода процессора за счет применения PCIe-v5 в IOD. Кристалл IOD поддерживает 12 контроллеров DDR5, Infinity Fabric, контроллеры SATA и памяти Compute Express Link (CXL) 1.1+. Для конкретного сервера можно сконфигурировать возможности их применения.

В кристалле рядом с контроллерами памяти находится также процессор AMD Secure, позволяя управлять рядом механизмов шифрования памяти, входящих в набор функций AMD Infinity Guard (см. об этом далее в описании средств безопасности Zen 4).

Кристалл IOD в EPYC Zen 4 имеет 12 IF-соединений с CCD. Эти CCD могут поддерживать одно или два IF-соединения к IOD. В моделях EPYC с четырьмя CCD можно использовать два таких соединения для оптимизации пропускной способности каждого CCD. Это имеет место в некоторых процессорах EPYC 9004 и всех процессорах EPYC 8004. В моделях процессоров с более чем четырьмя CCD, например в серии EPYC 9004, одно IF связывает каждый CCD с IOD.

Процессоры EPYC с большим разъемом SP5 имеют 128 линий PCIe и 12 каналов памяти. Процессоры с более простым SP6, где меньше контактов, поддерживают 96 линий PCIe и 6 каналов памяти.

Но в 2P-конфигурации сервера часть IOD должна работать на IF-соединение между процессорами, поэтому 2P-конфигурация EPYC 9004 на 2 процессора может предоставлять только до 160 линий PCIe.

Часть линий PCIe может быть настроена как встроенные контроллеры SATA, до 64 линий PCIe в EPYC 9004 (до 48 линий – в EPYC 8004) могут быть настроены для поддержки CXL 1.1+ для расширения памяти с когерентным кэшем и поддержки постоянной (persistent) памяти. В конструкции конкретной модели сервера бонусные линии PCIe-3.0 (см. рисунок 8) часто применяются для работы с не требующими высокой производительности устройствами ввода-вывода (например, с дисками SSD M.2), применяемыми для загрузки системы.

Но более важным для задач HPC представляется использование IF вместо части линий PCIe. Поскольку, как отмечено выше, IF совместимо с PCIe PHY, на физическом уровне часть линий PCIe могут применяться для IF. Один канал IF использует физическое соединение ×16 PCIe (это соответствует пропускная способность 128 ГБ/с для двунаправленной передачи). С процессорами EPYC 9004 можно создавать 2P-серверы, и тогда до четырех каналов IF (AMD называет их G-links, или xGMI, а что

такое GMI – см. ниже) могут применяться для связи между процессорами в сервере, что соответственно дает теоретическую двунаправленную пропускную способность до 512 ГБ/с. Это немного больше теоретической пропускной способности 12 поддерживаемых в IOD каналов DDR5-4800 (12×4.8 ГГц ×8 байт = 460.8 ГБ/с). Соответственно скорость доступа к памяти другого процессора близка к скорости локальной памяти [**23**].

Подобно тому, как 16 линий IF (PCIe PHY на нижнем уровне) объединяются в G-канал, PCIe-линии объединяются в P-канал (см. ниже рисунок 11). Для связи между процессорами можно использовать не четыре, а три G-канала – и тогда освободившиеся линии предоставляются как дополнительные линии PCIe-v5 (или могут применяться для CXL) [**121**] (о межпроцессорной связи см. в разделе 2.3).

Для связи между набором кристаллов CCD и IOD может использоваться 12 интерфейсов IF, именуемых GMI (Global Memory Interface). IOD могут работать с 4-, 8- или 12-ядерными CCD с ядрами Zen 4 или Zen 4c. Модели EPYC, содержащие не более чем четыре CCD, могут использовать для связи CCD с IOD по 2 GMI.

Кроме того, в состав IOD входит процессор безопасности, обеспечивающий, в частности, безопасное шифрование памяти (Secure Memory Encryption, SME) и безопасную зашифрованную виртуализацию (Secure Encrypted Virtualization, SEV), хаб контроллеров шин USB и др. Краткая информация о функциях безопасности EPYC Zen 4 приведена в данном разделе ниже.

Хотя обсуждение IOD начиналось с иерархии памяти, до сих пор собственно основная память практически не обсуждалась. Выше уже не раз упоминалось про наличие 12 каналов DDR5-4800 в EPYC 9004, дающих соответственно пиковую пропускную способность 460.8 ГБ/с (6 каналов в EPYC 8004 дают пропускную способность в два раза меньше). Благодаря применению DDR5 пропускная способность в EPYC 9004 стала раза в два больше, чем в соответствующих моделях EPYC Zen 3.

Для каждого канала памяти в IOD имеется свой контроллер UMC (Unified Memory Controller). Каждый UMC позволяет использовать один или два модуля DIMM на канал (сокращенно 1DPC или 2DPC), и соответственно до 24 DIMM на разъем при максимально допустимой общей емкости 6 ТБ [**121**] (типовой вариант для создания такой емкости – использование 3DS RDIMM емкостью по 256 ГБ).

Здесь полезно напомнить, что хотя в DDR5 поддерживается два слота DIMM на канал, такая их установка на одном канале вызывает уменьшение частоты и соответственно пропускной способности канала. Для борьбы с этим, как известно, и появилась буферизованная память, в том числе LRDIMM (см., например, [**129**]). Буферизованная память могла применяться с AMD EPYC Zen 3 [**103**], и поддерживается Zen 4 [**104**].

Наличие в IOD поддержки CXL 1.1+ дает возможность подсоединять еще CXL-память как промежуточный уровень в иерархии памяти между оперативной и внешней памятью. Это представляется наиболее актуальным для работы с базами данных в памяти и машинного обучения, хотя CXL-память может быть целесообразно применять и в других областях [**130**]. Предполагается эффективность применения CXL-памяти и для задач HPC [**131**].

Несколько более позднее обсуждение в обзоре памяти для Zen 4, несмотря на ее высокую важность для производительности, было сделано преднамеренно, поскольку для EPYC Zen 4 имеется много более тонких нюансов, относящихся к NUMA-особенностям памяти, для чего требуется отдельный анализ.

NUMA-особенности памяти. При использовании многокристальной технологии в процессоре будут разные задержки памяти в зависимости от варианта соединения отдельного кристалла (CCD в случае с EPYC) и контроллера памяти, т. е. будет NUMA. Последующая информация относительно NUMA базируется на [**121**], и относится к EPYC 9004.

Благодаря использованию расположенных в IOD контроллеров памяти, которое было реализовано еще в AMD Zen 2, неравномерность задержек памяти была существенно уменьшена. Последующие увеличения емкости кэша L3 в новых поколениях Zen также в определенном смысле элиминируют разницу задержек. Применение в Zen 4 новой версии IF по сравнению с использовавшейся в Zen 3 также способствует уменьшению этой разницы [**23**]. Однако для приложений, в которых неравномерность задержки памяти остается важной, возможна дополнительная оптимизация с явным учетом возможного разделения процессора на домены (узлы) NUMA, обозначаемые NPS (Nodes Per Socket).

Как видно на рисунке 9, более важные для целей данного обзора процессоры AMD, EPYC Genoa и Genoa-X, а также ориентированные на вычислительные системы с высокой плотностью упаковки процессоры Bergamo, имеют четыре «группы» кристаллов CCD, чему соответствуют четыре квадранта в процессоре. Из них можно образовать четыре узла NPS (см. рисунок 10).

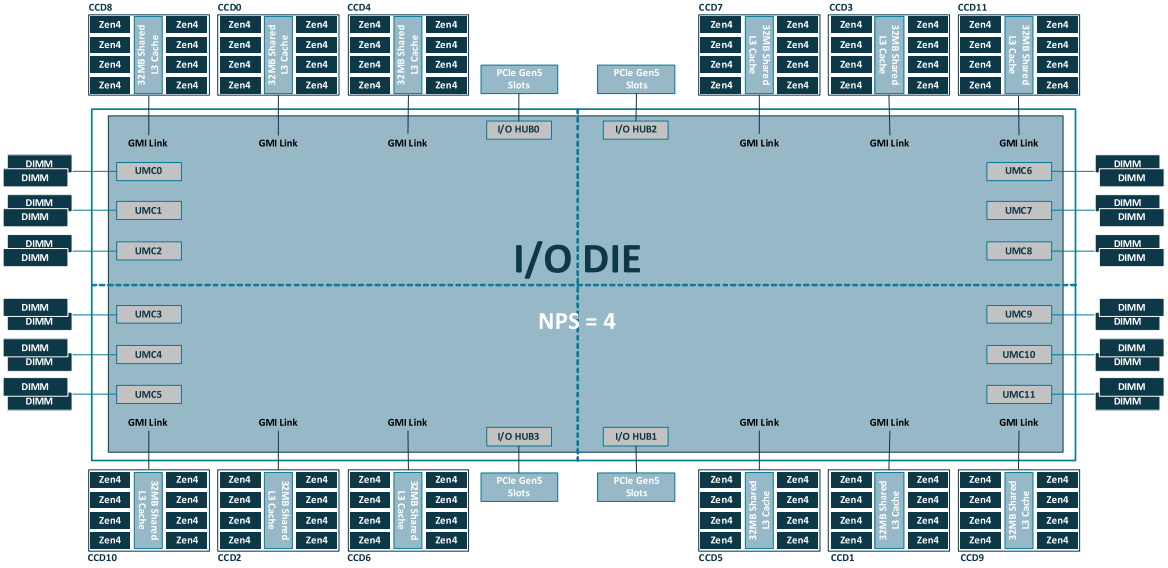


Рисунок 10. EPYC 9004 Genoa/Genoa-X – четыре квадранта и четыре NPS (рисунок из [121])

При NPS=4 процессор разделяется на 4 узла NPS, при NPS=2 – на 2 узла, при NPS=1 процессор имеет один общий NUMA-узел на сокет. Значения NPS устанавливаются в BIOS. Узлы NUMA обеспечивают локализацию ядер, памяти, хабов и устройств ввода-вывода. Можно использовать также и локализацию кэша L3 в каждом CCD, и тогда один процессор с 12 CCD может иметь до 12 узлов NUMA (подробнее см. [**121**]).

Здесь важно отметить чередование каналов памяти, используемое для возможности одновременной работы нескольких каналов памяти благодаря разбиению адресного пространства памяти на блоки. При NPS=1 чередуются все каналы памяти процессора, при NPS=2 в каждом узле чередуется по 6 каналов памяти, при NPS=4 чередование проводится свое в каждом квадранте. NUMA в 2P-серверах рассматривается далее в разделе 2.3 про вычислительные системы на базе Zen 4.

Получение информации о NUMA-конфигурации системы и привязка запускаемой программы к ядру процессора или узлу NUMA возможны в Linux с помощью команды numactl.

Что касается упомянутой выше CXL-памяти (имеется в виду не постоянная память, а поддерживающая обычные операции загрузки/сохранения), то она может восприниматься просто как отдельный домен NUMA-памяти, увеличивающий величину NPS на единицу (просто как еще один NUMA-узел без ядер) – см., например, [**130**].

TDP и регулировка частоты в EPYC 9004. Для задач HPC и ИИ, на что в большей степени и ориентирован обзор, актуальны именно процессоры этой серии.

Обычные значения расчетной тепловой мощности (TDP), «по умолчанию» применяемые в моделях EPYC Zen 4, приведены выше в таблице 6. BIOS поддерживает и режим настраиваемого TDP (cTDP), диапазон возможных значений которого известен для каждой конкретной модели Zen 4. cTDP могут использовать, например, производители серверов для оптимизации соответствующих им целевым установкам.

Например, TDP 96-ядерного AMD EPYC 9654 составляет 360 Вт, а cTDP – 320–400 Вт. Для задач HPC часто настраивают cTDP на максимальную мощность 400 Вт чтобы добиться максимальной производительности.

В BIOS у EPYC имеется важный режим детерминизма, который может быть установлен в Power или Performance. Этот режим использует уже упоминавшийся выше блок управления системой SMU для определения эффективной частоты каждого ядра. SMU отслеживает в режиме реального времени мощность, потребление тока и температуру во всех частях процессора и использует информацию о cTDP [**121**].

Выбор Performance уменьшает возможные различия производительности процессоров в кластере, используя общую базовую эталонную настройку для всех одинаковых моделей процессоров. Выбор Power использует преимущества возможной вариабельности «выработки» и может обеспечить улучшенную производительность во время выполнения. Это позволяет каждому процессору потреблять мощность в соответствии со своими индивидуальными возможностями, не допуская при этом превышения предела мощности cTDP. Выбор Power нужен для установки cTDP на максимальное значение [**121**].

В сочетании с настройками в BIOS BOOST=ON и детерминизма Power это позволяет SMU использовать всю возможную мощность, и может дать более высокую эффективную частоту выше базовой и для тяжелых SIMD-нагрузок (например, для DGEMM) [**121**].

Информация о поддерживаемых в EPYC 9004 соответствующих регулировке частоты состояниях процессоров и о регуляторах CPUfreq имеется в [**121**], на чем и основано изложение ниже. У всего процессора или его отдельных ядер поддерживаются состояния C0, C1, C2.

Для высокопроизводительных сетей с низкой задержкой (например, Infiniband), как указано в [**121**], состояние C2 в Linux нужно отключать. Настройки подобного типа выполняются с помощью команды cpupower для отдельных процессорных ядер или всего процессора. При включении `SMT` такая регулировка с помощью cpupower относится к «логическим процессорам», и отключать C2 надо для соответствующей пары логических процессоров.

В EPYC 9004 для работающего состояния C0 поддерживаются также состояния P0, P1, P2 с разными частотами. В boost-режиме (его можно включить в BIOS, а далее включать/отключать в Linux) каждое ядро может работать на частоте выше той, которая определяется состоянием P0.

Процессоры EPYC поддерживают четыре регулятора CPUfreq. Для HPC обычно используется регулятор performance, который выбирает частоту, устанавливаемую в P0. При включенном boost-режиме регулятор пытается поднять частоту до максимального boost-значения [**121**].

Регулятор ondemand устанавливает частоту в зависимости от используемой нагрузки. Это способствует быстрому выходу на максимальную рабочую частоту с последующим пошаговым снижением до P2 при простаивании. Это плохо для «коротко-живущих» нитей [**121**].

Регулятор conservative аналогичен ondemand, но предлагает более плавный переход к максимальной частоте и быстрый возврат к P2 на холостом ходу. Наконец, powersave устанавливает самую низкую поддерживаемую частоту, фиксируя ее на уровне P2.

Выбор исполняемого регулятора также осуществляется в Linux с помощью cpupower.

Аппаратная поддержка виртуализации. Как уже было указано выше, про эти особенности Zen 4 здесь будут сказано предельно кратко. Технология виртуализации AMD-V применяется в процессорах фирмы очень давно, и для нее используется соответствующая часть ISA. AMD-V включает в себя и такие функции, как виртуализацию ввода-вывода (AMD-Vi), поддерживающую прямое назначение устройств виртуальным машинам, и виртуализацию прерываний AVIC (Advanced Virtual Interrupt Controller).

Но самым важным здесь представляется минимизация накладных расходов на виртуализацию виртуальной памяти. AMD использует для этого вложенные таблицы страниц, применяя трансляцию адресов второго уровня (SLAT), позволяющую избежать накладных расходов при использовании теневых страниц. AMD для этого давно применяет свою технологию Rapid Virtualization Indexing (RVI).

Учитывая возможности расширения памяти с помощью контроллеров CXL для адреса виртуальной памяти в Zen 4 используется уже 57 бит, и реализован пятый уровень вложенных таблиц страниц [**23**] (пятый уровень таблицы страниц для работы с 57-битными адресами был предложен Intel в 2017 году [**132**]).

При использовании виртуализации резко возрастают требования к безопасности, поскольку необходимо обеспечить изоляцию данных в разных виртуальных машинах. Для часто используемого второго уровня вложенных страниц, SNP, у AMD давно используются и средства обеспечения безопасности SEV-SNP (SEV – Secure Encripted Virtualization), см. ниже.

Прозрачные огромные страницы (Transparent Huge Pages, THP), используемые в Linux для уменьшения промахов в буфере TLB для компьютеров с большими объемами памяти, в Zen 4 могут иметь размер 2 МБ [**121**]. THP именуются прозрачными, поскольку при включении работы с ними в ядре Linux приложения об этом явно не уведомляются – в предположении повышения их производительности за счет сокращения накладных расходов на управление памятью при использовании TLB

благодаря сокращению числа требуемых записей в этом буфере. Включение поддержки THP в ядре Linux может повысить работу HPC-приложений, но при работе с БД чаще уменьшает производительность, и его там соответственно отключают.

Безопасность. AMD уделяет большое внимание обеспечению безопасности в Zen 4. Естественно, эти средства безопасности поэтапно развивались, начиная с первого поколения Zen. AMD принимает меры по устранению уязвимостей, известных во время разработки, и это не приводит к необходимости изменять прикладные программы [**23**].

Расширение применения виртуализации, естественной для многоядерных Zen 4, как отмечено выше, требует повышенной изоляции виртуальных машин, и сопровождалось соответственно усилением аппаратной поддержки средств безопасности. Поскольку у AMD имеется целый ряд таких средств, для некоторых имеется свое собственное описание, мы ограничимся здесь в основном указанием ссылок на них. Эти средства — Infinity Guard (в нем интегрированы упоминавшиеся выше средства SME и SEV-SNP) [**133**], безопасности памяти [**134**], теневой стек (Shadow Stack) и защита боковых каналов (Side Channels).

Большинство потенциальных проблем с безопасностью имеются у всех современных процессоров. Теневой стек и защита боковых каналов имеют прямое отношение к задачам виртуализации [**135**]: нужна строгая изоляция между различными виртуализированными гостями, и каждой виртуальной машине нужен собственный ключ шифрования. Обеспечиваются и безопасные вложенные таблицы страниц (nested paging). В состав IOD входит собственный процессор безопасности; поддерживается также многоключевое шифрование (SMKE) [**133**], которое позволяет гипервизорам выборочно шифровать диапазоны адресного пространства в памяти, подключенной к CXL [**23**].

Публикации по средствам обеспечения безопасности в Zen 4, вероятно, будут скоро появляться. Здесь мы укажем на работу [**136**], в которой анализируются доверенные платформенные модули и возможности процессора безопасности AMD в Zen 3. Пока работы по изучению безопасности проводятся еще и для AMD Zen 2 [**137**, **138**].

Особенности процессоров EPYC серии 4004. Эти процессоры применяются в серверах с небольшим числом ядер и одним сокетом. Информация об EPYC 4004 имеется в майской версии 2024 года документа [**23**], на чем и основано краткое изложение в этом подразделе.

EPYC 4004 содержат от 4 до 16 ядер и до двух CCD. Максимально возможная емкость кэша L3 составляет соответственно 64 МБ, а 3D V-кэша – 128 МБ. В этих процессорах используется более простой IOD с меньшей, по сравнению с более старшими сериями процессоров, площадью.

В этом IOD по сравнению с другими сериями EPYC Zen 4 имеется ряд модификаций. В частности, добавлен графический процессор AMD с архитектурой RDNA2. В IOD здесь имеется одно устройство SerDes с 16 линиями, и одно – с 12 линиями, и это дает до 28 линий PCIe-v5. Имеется два канала памяти DDR5-5200 с общей теоретической пропускной способностью 83.2 ГБ/с, а емкость этой памяти может достигать 192 ГБ.

Но благодаря использованию небольшого числа ядер они могут работать в процессоре на высоких тактовых частотах, повышая производительность каждого ядра. Так, 16-ядерная модель EPYC 4564P имеет базовую частоту 4.5 ГГц, а ускоренную – до 5.7 ГГц [**139**].

Процессор безопасности в IOD этой серии EPYC не поддерживает SEV и SMKE. EPYC 4004 ориентируются на применение в серверах для малого бизнеса, хостинга выделенных серверов и других областей, где достаточно применение процессоров с небольшим числом ядер.

2.3. О вычислительных системах на базе процессоров EPYC 9004

Процессоры EPYC 8004 и 4004 могут использоваться только в компьютерах с одним сокетом и далее не рассматриваются. Реально на месте этого раздела можно было бы обсуждать все от материнских плат до суперкомпьютеров, поскольку ранее анализировались только процессоры. Но в соответствии с направленностью обзора рассмотрение ниже ограничено некоторыми относящимися к EPYC 9004 особенностями – а соответствующие материнские платы и серверы выпускают все ведущие производители. Имеются самые разные варианты серверов и систем из нескольких модулей-серверов и с воздушным, и с водяным охлаждением, как без, так и с GPU.

С EPYC 9004 могут создаваться 1P- и 2P-серверы. В последних, как было отмечено выше, для связи между процессорами используются три или четыре канала IF (G-канала, см. рисунок 11). Для серверов, требующих интенсивного ввода-вывода, можно использовать три G-канала, тогда освободившийся канал можно выделить для PCIe-v5, получая в сумме 160 линий PCIe на весь сервер.

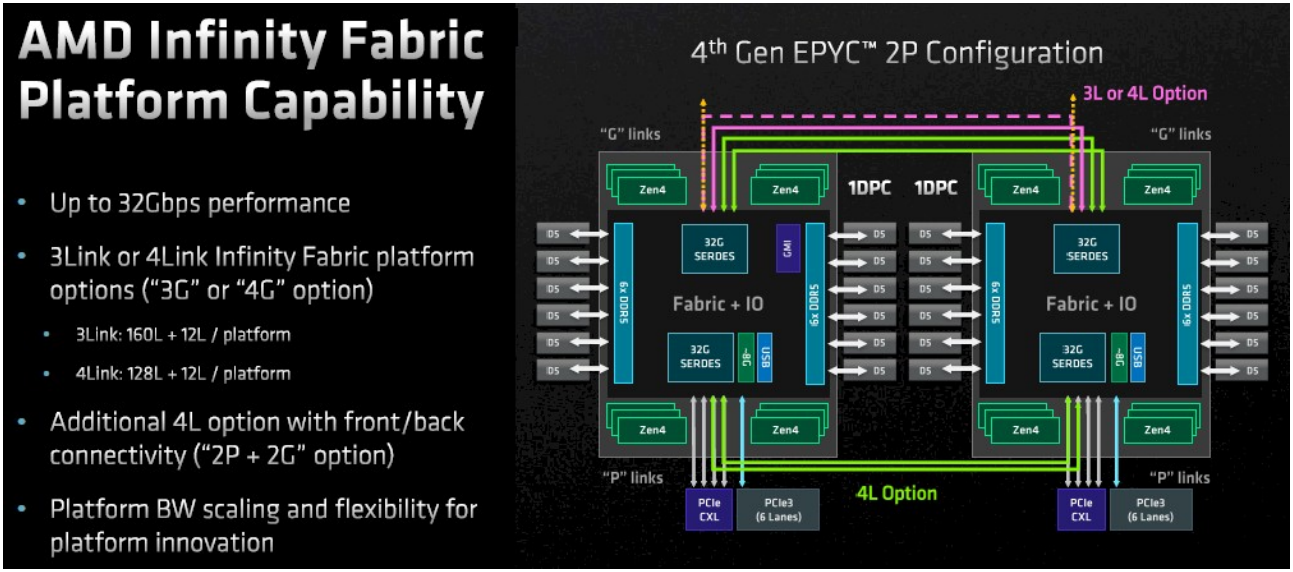


Рисунок 11. 2P-конфигурация с EPYC 9004 (рисунок из [**127**])

Ориентация AMD на HPC видна даже из руководства по BIOS. Параметры BIOS можно отслеживать в Linux, а некоторые даже изменять. В BIOS серверов с EPYC 9004 есть целый ряд интересных опций, способствующих возможному улучшению производительности или энергоэффективности. Можно с шагом 1 уменьшать количество используемых ядер в CCD от 8 до 1, сохраняя при этом неизменной общую емкость кэша L3. Можно уменьшать количество активных CCD в процессоре, сохраняя при этом количество ядер на CCD [**121**].

Как уже было указано в разделе 2.2, в BIOS есть опция, разрешающая или запрещающая увеличение используемой частоты процессора. Если она включена, то в командной строке Linux ее можно отключать. Понятно, что есть опция включения/отключения SMT (каждое ядро может поддерживать одну или две нити). Есть опция, повышающая в Linux эффективность обработки прерываний в конфигурации с большим числом процессорных ядер.

Можно явно устанавливать скорость памяти (до 5400 миллионов передач в секунду, что говорит о возможной работе Zen4 и с DDR5-5400). Система с двумя сокетами имеет 24 канала памяти. При установке в BIOS NPS=0 подавляется использование NUMA, и вся общая память, подсоединенная к обоим процессорам сервера, выглядит как однородная. Память при этом чередуется по всем каналам в едином адресном пространстве сервера. Чередование памяти в BIOS можно отключить [**121**].

Ряд опций относится к работе с xGMI. Подробнее про работу с xGMI и опции BIOS, а также другие важные для HPC особенности, включая работу с межсоединениями Mellanox, см. в [**121**].

Аналогично руководству [**121**] для HPC, AMD предлагает руководство для эффективной работы с ИИ [**140**], где описано, например, как эффективно создавать ориентированные на обработку больших данных Hadoop-кластеры.

С точки зрения безопасности серверов на базе EPYC 9004 укажем, что они получают сертификат FIPS 140-3 (например, сервер от Lenovo [**141**]).

В завершение раздела про вычислительные системы с EPYC 9004 проиллюстрируем их примерами серверов с этими процессорами, и использующих их узлов суперкомпьютеров из TOP500.

В качестве типичного варианта построения такого сервера можно указать на детальную блок-схему для Lenovo ThinkSystem SR665 V3, относящегося к традиционным 2P-серверам с формфактором 2U [**142**] (см. рисунок 12).

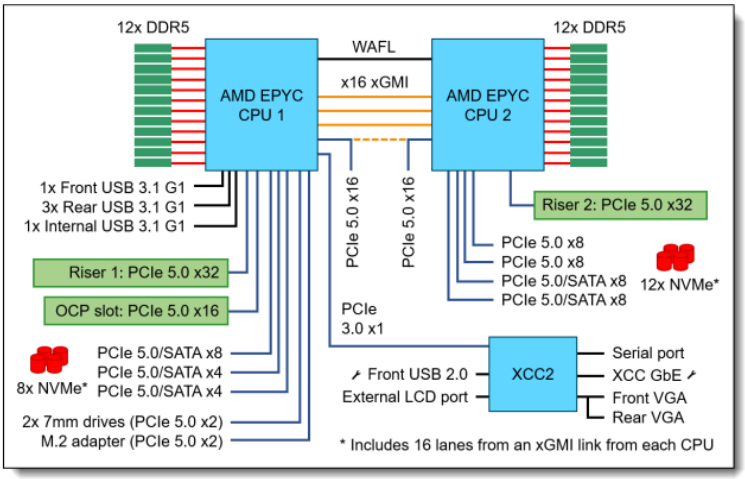


Рисунок 12. Блок-схема архитектуры сервера Lenovo SR665 V3 (рисунок из [142])

В качестве примера гомогенных узлов суперкомпьютеров укажем на Shaheen III в Саудовской Аравии, появившийся в 2023 году и занимающий 23-е место в июньском списке TOP500 2024 года. Там узлы содержат по два процессора EPYC 9654 [143].

В качестве примера для гетерогенных узлов с GPU нельзя не указать на первый в мире экзафлопсный суперкомпьютер Frontier. Хотя там в узлах используются 64-ядерные EPYC 7A53 с ядрами Milan (по одному процессору на узел), но в процессоре уже применяется более современный IOD [144]. Про лидирующий начиная с ноябрьского списка TOP500 2024 года суперкомпьютер El Capitan, в гетерогенных узлах которого применяются процессоры Zen 4, говорится в разделе 7.

Другим интересным примером суперкомпьютера из июньского (2024 года) списка TOP500 с GPU Nvidia H100 является немецкий суперкомпьютер HoreKa-Teal, занимающий 6-е место в соответствующем списке GREEN 500. В узлах HoreKa-Teal используются серверы Lenovo ThinkSystem SD665-N V3, содержащие по два 32-ядерных EPYC 9354 [145].

2.4. О производительности вычислительных систем с EPYC Zen 4

Здесь будут рассмотрены данные о производительности в первую очередь серверов с процессорами EPYC Zen 4 серии 9004, в том числе в сравнении с серверными процессорами x86 предыдущего поколения и

ARM-процессорами, хотя в некоторых случаях проводится и полезное сравнение с GPU. Данные о производительности кластеров приводятся исключительно для демонстрации масштабирования в них производительности конкретных приложений.

Рассматривая в обзоре сопоставления производительности разных процессоров, мы в первую очередь обращаем внимание на сопоставления старших моделей (которые часто применяются в суперкомпьютерах), а также на сопоставления моделей, имеющих одинаковое число процессорных ядер. Отдельно мы обращаем определенное внимание также и на модели среднего класса, которые активно используются в HPC, и здесь наиболее интересно именно сравнение моделей с одинаковым числом ядер.

Понятно, что интересно также сопоставление с близкими по ценам моделями или с близкими показателями TDP, но это считается здесь отчасти второстепенным. Кроме того, интерес представляет даже сопоставление производительности моделей с разным числом процессорных ядер, так как это может давать и совсем грубые начальные предположения о возможном масштабировании производительности с числом ядер.

Что касается роста производительности EPYC Zen 4 по сравнению с Zen 3 – то он очевиден из-за увеличения числа ядер, их тактовых частот, появления поддержки AVX-512, роста пропускной способности используемой памяти, емкости 3D-кэша L3 и др. Понятно, что это дает и увеличение производительности моделей Zen 4 при одинаковом числе ядер с Zen 3.

Имеется огромное количество данных о производительности EPYC Zen 4 на сайтах широко известных массовых тестов производительности spec.org, openbenchmarking.org, на сайте AMD (в первую очередь на хабе документации [**146**]) или на других сайтах, ориентированных, например, на конкретные области применения. Они демонстрируют повышение производительности Zen 4 относительно Zen 3, и обычно – преимущества в производительности относительно Xeon SPR и EMR.

В обзоре приводится лишь небольшая часть таких данных, относящаяся в первую очередь к HPC, и выбранная в качестве более актуальных. В качестве общей иллюстрации ускорения производительности известных HPC-приложений в старшей модели Zen 4 Genoa (EPYC 9654) по сравнению со старшей моделью Zen 3 (EPYC 7763) укажем на данные из доклада AMD на семинаре в Academia Sinica (Национальной академии Китайской Республики Тайвань) [**89**], см. рисунок 13.

Правда, можно предположить, что эти числовые данные (в процентах) о приросте производительности не средние, и их скорее разумно помечать

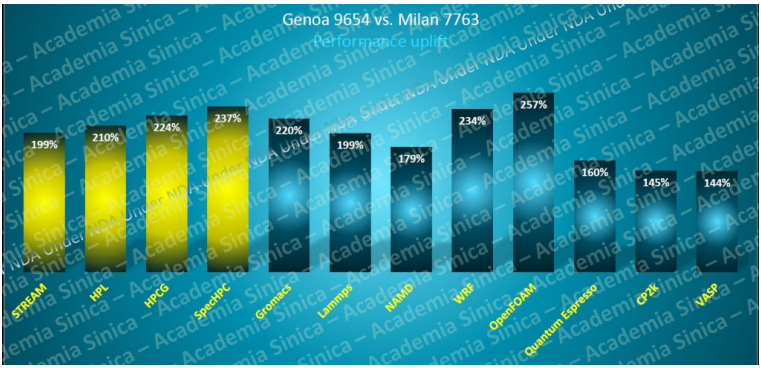


Рисунок 13. Относительная производительность EPYC 9654 по сравнению с EPYC 7763 на тестах и приложениях (рисунок из [89])

признаком «вплоть до». Большое количество данных о производительности разных моделей Zen 3 и Zen 4 для указанных на этом рисунке приложениях и тестах, как и для многих других, можно найти в поиске на сайте openbenchmarking по имени теста, что предложит URL типа `https://openbenchmarking.org/test/pts/cp2k-1.3.0`, где cp2k – возможное имя приложения (или теста), а через тире – номер версии.

Данные о производительности для 1P- и 2P-конфигурации сервера с моделями EPYC 9654 и 9554 на большом количестве разных приложений, в том числе показывающих преимущества в производительности относительно EPYC Zen 3 и 3-го поколения масштабируемых процессоров Xeon, имеются в [147]. Сопоставление производительности EPYC 9004 с Xeon EMR и Xeon SPR/Xeon Max в основном проводится далее в разделах 4.1–4.4.

В качестве другого примера, охватывающего данные о производительности по большому числу самых разных областей применения, включая многие, не охваченные в данном обзоре, сошлемся на [148]. AMD EPYC стали очень активно использоваться для тестов PTS, там для 2P-сервера с EPYC 9654 проведены даже сопоставления достигаемой производительности при использовании разных современных дистрибутивов Linux без их любых модификаций на широком наборе тестов, включая молекулярную динамику (NAMD), работу с базами данных (PostgreSQL и MariaDB), задачи ИИ (в том числе тесты OpenVINO и OneDNN) и много других [149].

AMD приводит много данных по производительности EPYC 9004, из них мы отобрали только кажущиеся наиболее актуальными. Но перед

обсуждением конкретных данных тестов производительности, включая данные тестов производительности приложений, полезно указать на общую информацию об ускорении, достигаемом за счет применения поддержки AVX-512 в Zen 4, которая в [**150**] была получена при тестировании 2P-сервера с EPYC 9654. Там продемонстрировано существенное ускорение на многих тестах для задач ИИ, включая TensorFlow (с моделями AlexNet, ResNet-50 и GoogleNet), OpenVINO, некоторых тестов oneDNN и др. Ускорение EPYC 9004 в задачах ИИ было ожидаемо из-за поддержки в реализованном в Zen 4 расширении AVX-512 возможностей VNNI и формата BF16.

Заметный прирост производительности при использовании AVX-512 был получен в [**150**] в мини-приложении молекулярного докинга miniBUDE. Это, возможно, связано с применением здесь трехмерных матричных преобразований [**151**]. В других HPC-приложениях, не использующих традиционные умножения плотных матриц, в том числе в приложении CP2K для молекулярной динамики или в средствах OpenFOAM для CFD, прирост производительности при включении поддержки AVX-512 в [**150**] был мал.

2.4.1. *Данные о производительности в различных тестах SPEC*

Естественно начать анализ с данных различных тестов SPECcpu 2017. Соответствующие результаты для производительности с плавающей запятой приведены далее в таблице 27 (там приведены данные этого теста не только для EPYC 9004, но и для Xeon SPR/Xeon Max и Xeon EMR). Много данных, показывающих преимущество в производительности в тестах SPECcpu 2017, SPEComp 2012, SPECmpi 2007 и SPEChpc 2021 разных моделей EPYC 9004 старшего и среднего класса по сравнению с соответствующими моделями EPYC Zen 3 и Xeon ICL (третьего поколения масштабируемых процессоров), в том числе при сравнении с одинаковым числом процессорных ядер, приведены в документе AMD для HPC [**152**].

Поскольку преимущества в производительности EPYC 9004 по сравнению с процессорами EPYC Zen 3 достаточно очевидны, наш анализ данных тестов SPECcpu 2017 сосредоточен исключительно на сопоставлении производительности с Xeon SPR, EMR и Xeon Max, и проводится далее в отдельных разделах 4.1 и 5.4. По аналогичным причинам данные тестов SPEChpc 2021 в таблице 28 также приведены в разделе 4.1, в котором в основном и сконцентрировано сопоставление производительности EPYC 9004 с масштабируемыми процессорами Xeon 4-го и 5-го поколений.

Что касается тестов SPEComp 2012, то соответствующие результаты из [**153**] приведены в таблице 7.

Таблица 7. Данные о производительности рассматриваемых процессоров x86 в тесте SPEComp 2012

Модели	Число ЦП	Число ядер	Результат SPEComp_2012	
			Пиковый	Базовый
EPYC 9654	1	96	48.4[1]	47.5[1]
	2	192	86.0[1]	83.0[1]
EPYC 9754	1	128	64.2[2]	61.7[2]
	2	256	108[2]	104[2]
Xeon 8480+	2	112	—	61.0[3]
Xeon 8490H	2	120	—	61.3[3]
	4	240	108[4]	100[4]
	8	480	138[5]	126[5]
Xeon 8592+	2	128	—	62.0

Приведены максимальные достигнутые данные на 1.09.2024.

Отправители результата и использованные серверы:

[1] Lenovo ThinkSystem SR655 V3;
[2] Cisco UCS C245 M8 SM;
[3] xFusion FusionServer 2288H V7;
[4] Lenovo ThinkSystem SR860 V3;
[5] Lenovo ThinkSystem SR950V3.

Во всех представленных в таблице серверах, в которых проводились тесты, число процессорных ядер достаточно сильно отличается друг от друга, и большая производительность связана просто с большим числом ядер.

В приведенных в этой таблице данных использовались возможности SMT в процессорах, и применялось по 2 нити на физическое ядро. Данные этой таблицы при переходе от 1P- к 2P-серверам, и от двух- к четырехпроцессорным серверам показывают, какая может быть масштабируемость производительности в этих тестах с ростом числа ядер и соответственно числа используемых нитей OpenMP.

Производительность одного 128-ядерного EPYC 9754 (Bergamo) лишь немного выше, чем в 2P-сервере с Xeon 8480+ (со 112 ядрами на сервер) для базового (без индивидуальной оптимизации для каждого компонента теста) результата, и немного ниже, чем в 2P-сервере с Xeon 8592+ (со 128 ядрами на сервер). Однако процессоры Bergamo исходно не ориентированы на HPC-область и основаны на более медленных ядрах по сравнению с EPYC Genoa, а 2P-сервер с EPYC Genoa (с 96-ядерными моделями EPYC 9654), естественно, в этом тесте 2P-серверы с Xeon опережает.

Вторым важным моментом здесь является пропускная способность памяти. Когда общее число всех ядер в сопоставляемых серверах близко, 2P-система получает преимущество, так как каждый из процессоров имеет свои контроллеры памяти, и суммарная пропускная способность памяти удваивается. Поэтому тот факт, что производительность сервера с одним 128-ядерным процессором Bergamo EPYC 9754 очень немного уступает 2P-серверу с двумя топ-процессорами пятого поколения Xeon 8592+ с тем же общим числом в 128 ядер, говорит о сочетании высокой конкурентоспособности по производительности отдельных ядер Bergamo и хорошего ее масштабирования (а у Genoa – тем более).

Тест SPEComp 2012 для HPC является интегральным (его компоненты задействуют разные области применения), и поэтому результаты весьма актуальны. Но здесь надо было бы исследовать, как масштабируется производительность с ростом числа используемых в расчете ядер.

В [**154**] приведены интересные данные, показывающие влияние на результат SPEComp числа используемых нитей при тестировании производительности EPYC 9654 со включенной поддержкой SMT. Производительность при переходе от одного ядра (2 нити) к двум ядрам (4 нити) возрастает всего на 25%, но продолжает увеличиваться вплоть до использования всех 96 ядер (192 нитей). На 96 ядрах достигнутое ускорение было более чем в 24 раза. Но с другой стороны это не соответствует известным данным о возможных прекращениях масштабирования производительности с увеличением до гораздо меньшего числа ядер, в том числе на приложениях CFD, и общим типовым рекомендациям по выбору процессоров для задач CFD [**155**]. При этом 5 из 18 тестов в составе SPEComp относятся к CFD, не считая еще и аэродинамический тест о прогнозе погоды (эти области являются связанными памятью). Такие ситуации с быстрым уменьшением роста производительности при увеличении числа использованных ядер часто имеют место и в других приложениях HPC, см. об этом далее.

Здесь также нужно обратить внимание на то, что в документе Lenovo [**154**], на серверах которой были получены некоторые наивысшие приведенные в этой таблице результаты SPEComp для EPYC 9004, показано, что включение режима SMT приводит к возрастанию производительности более чем в полтора раза. Это противоречит известным данным о слабом влиянии включения SMT на производительность в приложениях HPC и, в частности, рекомендации выключать SMT для CFD-приложений [**155**]. Но ясно, что исследование зависимости производительности от числа используемых ядер очень важно.

В таблице 8 представлены данные о производительности EPYC 9004 (все данные относятся к одному узлу кластера) в тестах SPECmpiM 2007. В ней приведены также данные для процессора EPYC 9755 (Zen 5, см. об этом в разделе 6).

Таблица 8. Производительность некоторых моделей AMD EPYC в тесте SPECmpiM 2007

Модель	Число процессоров	Число рангов	Число доступных нитей	Число ядер	Base	Peak
EPYC 9754[1]	1	128	256	128	36.4	36.4
EPYC 9654P[2]	1	96	96	96	36.4	36.4
EPYC 9654[3]	1	96	192	96	36.0	36.0
EPYC 9654[4]	2	192	384	192	64.1	64.1
EPYC 9654[5]	2	192	192	96	65.4	65.4
EPYC 9755[6]	2	256	256	256	96.6	96.6

Отправители данных и серверы, на которых проводился тест:
[1] Supermicro A+ Server 1025CS-TNR;
[2] Lenovo ThinkSystem SR655 V3;
[3] Supermicro A+ Server 1115CS-TNR;
[4] Supermicro A+ Server 2125HS-TNR;
[5] Lenovo ThinkSystem SR665 V3;
[6] Supermicro Hyper A+ Server AS -2126HS-TN.
Данные из [**156**] на 5.12.2024.

Соответствующие данные для обсуждаемых в обзоре процессоров Xeon SPR, Xeon EMR и Xeon Max отсутствуют, а 2P-сервер с 56-ядерными Xeon 3-го поколения (8380H) показал результат 40.4 (одинаковые базовое и пиковое значения) – чуть больше, чем 1P-сервер с EPYC 9654.

Характеристики производительности EPYC 9004 для Java-приложений и соответствующих серверных рабочих нагрузок в тестах SPECjbb2015 приведены в таблице 9. В данных этой таблицы отобраны только те процессоры, которые могут представлять наибольший интерес с точки зрения автора.

Таблица 9. Данные о производительности EPYC и Xeon в тестах SPECjbb2015

Модель	Число ЦП	composite		MultiJVM		Distributed	
		max_jOPS	critical_jOPS	max_jOPS	critical_jOPS	max_jOPS	critical_jOPS
EPYC 9754	1P	405933	379972[1]	**446246**	192284	**441549**	195109
				362000	**335166**[1]	366259	**349986**[1]
	2P	**645699**	538879	**892492**	401899[11]	**888595**	320876
		632433	**578007**[2]	703744	**648506**[2]	706963	**659783**[2]
EPYC 9684X	1P	**442094**	**414785**[1]	**494311**	218746	**497170**	222613
				383203	**357304**[1]	388313	**357576**[1]
	2P	**628244**	566406	**988621**	410550[2]	**970665**	387180[2]
		621185	**571538**[3]	741034	**715483**[12]	745969	**717538**[12]
EPYC 9654	1P	**381145**	356013[1]	494311	218746[1]	**424600**	190451
		376911	**356425**[4]			344964	**327561**[1]
	2P	607067	556933[5]	**967587**	373830[13]	**838821**	361028
				741034	**715483**[12]	672893	**635876**[2]
Xeon 8592+	1P	—	—	258368	152750[14]	—	—
	2P	-	-	558626	310911[15]	-	-
				480007	421207[8]		
Xeon 8580	2P	391485	155366[6]	464765	254278[6]	—	—
Xeon 8570	2P	372676	147612[6]	451099	233491[6]	—	—
Xeon 8558U	1P	—	—	203534	112446[16]	—	—
Xeon 8490H	1P	200721	124502[7]	213001	125650[17]	—	—
	2P	421768	261161[8]	505379	257205[18]	-	-
				458295	368979[8]		
	4P	428829	299584[9]	884811	472868[19]	**814136**	322485
				733918	618248[9]	743435	**624702**[9]
	8P	—	—	1356786	1012835[20]	1127931	896441[20]
Xeon 8480+	2P	343031	309274	**476987**	243526[21]	**421268**	213428[22]
		338796	**309284**[10]	371890	**313390**[10]	373578	**309482**[10]
Xeon 8470	2P	**334561**	226875	**402335**	210668	**402335**	207396
		317651	**297873**[10]	363716	**299820**[10]	359630	**298159**[10]

Данные из [https://spec.org/jbb2015/results/ 29.08.2024]; приведены максимальные из полученных результаты на 29.08.2024. Отправители этих результатов и использованные системы:
[1] ASUS RS520A-E12-RS12U; [2] ASUS RS720A-E12-RS12; [3] Dell PowerEdge R7625; [4] Lenovo ThinkSystem SR655 V3; [5] ASUS RS700A-E12-RS12U; [6] Nettrix R620 G50; [7] Supermicro SuperServer SYS-521E-WR; [8] ASUS RS720-E11-RS12U; [9] Lenovo ThinkSystem SR860 V3; [10] Dell PowerEdge MX760c; [11] Kaytus KR1280V2(KR1280-E2-A0-R0-00); [12] Lenovo ThinkSystem SR665 V3; [13] Cisco UCS C245 M8; [14] Supermicro материнская плата X13SEI-TF; [15] H3C UniServer R4900 G6; [16] Supermicro SuperServer SYS-521C-NR; [17] IEIT I22G7; [18] xFusion FusionServer 5288 V7; [19] xFusion FusionServer 5885H V7; [20] Lenovo ThinkSystem SR950 V3; [21] Inspur NF5180M7; [22] Dell PowerEdge R760.

Результаты в каждой клетке относятся к тестам одного сервера.

Поскольку максимальный средний показатель critical_jOPS может достигаться в одном исполнении теста, а наивысший max_jOPS – в другом, во многих случаях максимальная производительность с применением одного и того же процессора отражается парой представленных в этой таблице результатов. Хотя всегда в верхней строчке из такой пары приводятся данные с максимальным max_jOPS, для удобства максимальный результат из пары max_jOPS, critical_jOPS отмечается жирным шрифтом.

Интересно, что во всех трех вариантах теста SPECjbb 2015 системы со 128-ядерными Bergamo EPYC 9754 с уменьшенными емкостями кэша и частотами ядер уступили системам с 96-ядерным Genoa EPYC 9684X (2P-системы уступили только в двух из трех вариантов), но обогнали системы с 96-ядерными Genoa EPYC 9654, не имеющими расширенного кэша L3 (за исключением варианта с MultiJVM).

Данные этих тестов для EPYC 9004 интересны также с точки зрения оценки производительности с процессорами Bergamo и для оценки влияния на производительность расширенного кэша L3.

Данные тестов энергоэффективности (производительности на Вт), SPECpower_ssj 2008 (см. таблицу 10) относятся к серверным бизнес-приложениям Java. Интересно, что разные ARM-процессоры (от Ampere) не показывают в этих тестах ожидаемых для ARM успехов, и отстают от x86 (не представленные в таблице данные для Ampere Altra Q64-30 давали еще более низкий результат).

Результаты этой таблицы демонстрируют сильное превосходство старших моделей EPYC 9004 над старшими моделями Xeon 4-го и 5-го поколений. Многократные «мировые рекорды» по SPECpower_ssj 2008 принадлежат разным серверам на базе ориентированных на энергоэффективность процессоров Bergamo (старшей модели EPYC 9754).

Максимальные показатели в этих тестах достигались на 2P-серверах. Дальнейшее повышение числа процессоров Xeon SPR в сервере (их может быть там до 8) приводит к уменьшению энергоэффективности.

Ниже в таблице 11 приведены данные, уже достаточно далеко отходящие от оценки производительности собственно процессора, но являющиеся интегральными показателями для широко используемых ЦОД. Это тесты производительности консолидированных виртуальных серверов SPECvirt_sc2013, в которых используются смешанные рабочие нагрузки – Web-серверов, mail-серверов, работы с БД и других широко используемых видов нагрузок.

Таблица 10. Данные по производительности на Вт разных процессоров в тестах SPECpower_ssj 2008

Модели процессоров	Число ЦП	Число ядер	Общее количество ssj_ops на ватт	Цена[14]
EPYC 9654	1	96	27448[1]	$11805
	2	192	30602[2]	
EPYC 9754	1	128	36637[3]	$11900
	2	256	37678[1]	
Ampere Altra Q80-30	1	80	12718[4]	$3950[15]
Ampere Altra Max M128-30	1	128	11497[5]	$5800[15]
	2	256	12195[6]	
Xeon 8592+	1	64	17224[7]	$11600
	2	128	20408[8]	
Xeon 8490H	1	60	14537[9]	$17000
	2	120	17415[10]	
	4	240	15078[12]	
	8	480	11898[12]	
Xeon 8480+	1	56	10290[13]	$10710
	2	112	16653[10]	

В таблице приведены максимальные достигнутые на 15.11.2024 показатели для приведенных процессоров из [https://spec.org/power_ssj2008/results/15.11.2024].
Владельцы аппаратуры и используемые серверы:
[1] Lenovo Think System SR655 V3;
[2] ASUS RS720A-E12-RS12;
[3] Lenovo Think System SD535 V3;
[4] FALINUX AnyStor-700EC-NM;
[5] Supermicro, GIGABYTE R152-P31;
[6] FOXCONN (FII), Mt. Collins;
[7] Lenovo ThinkSystem SD530 V3;
[9] Supermicro SuperServer SYS-521E-WR;
[8] ASUS SC4000-E11;
[10] xFusion FusionServer 2288H V7;
[11] Lenovo ThinkSystem SR860 V3;
[12] FUJITSU Server PRIMERGY RX8770 M7;
[13] Supermicro SuperServer SYS-521C-NR;
[14] Данные о ценах из [49] от 15.11.2024, за исключением цен для процессоров ARM
[15] Данные из [157].

Хотя сравнительных данных этих тестов не так много, как для других использованных в обзоре тестов SPEC, а результаты существенно зависят от отличавшихся показателей как в аппаратуре (например, емкости памяти), так и в программном обеспечении (например, гипервизора), полученные данные о производительности в целом соответствуют ожиданиям. Можно только отметить успех Xeon 8592+. Эти данные могут быть полезными и

Таблица 11. Данные о производительности процессоров в тестах SPECvirt

Процессор	Число ЦП	SPECvirt_sc2013VMs	Платформа
EPYC 9654	2P	8336462	China Mobile (Suzhou) Software Technology
Xeon 8592+	2P	9801546	xFusion Digital Technologies Co., Ltd.
Xeon 8490H	2P	7528420	xFusion Digital Technologies Co., Ltd
Xeon 8480+	2P	5277294	China Electronics Cloud Technology Co., Ltd.
Xeon 8280L	8P	11966672	Lenovo Global Technology

Данные из [https://spec.org/virt_sc2013/results/specvirt_sc2013_perf.html 11.02.2025]. Примечание. После приводится число виртуальных машин. Приведены максимально достигнутые показатели для серверов с процессорами приведенных типов на 29.08.2024.

вообще для оценки возможностей виртуализации.

Можно сказать, что во всех разных приведенных тестах SPEC старшие модели процессоров EPYC 9004 опережают по производительности Xeon SPR и EMR (за исключением некоторых данных для SPECvirt_sc2013).

2.4.2. *Тесты пропускной способности памяти stream*

Анализ производительности EPYC 9004 в тестах stream сразу после рассмотрения тестов SPEC мы начинаем вследствие все возрастающего числа ситуаций, когда приложения оказываются связанными памятью. Сам рост пропускной способности памяти при переходе от EPYC Zen 3 к Zen 4 очевиден просто хотя бы из-за перехода от применения DDR4-3200 в Zen 3 Milan на DDR5-4800 в EPYC Zen 4. Пропускная способность в stream triad при замене 2P-сервера с 64-ядерными EPYC 7773X или EPYC 7763 на 2P-сервер с 96-ядерными EPYC 9654 возрастает в два раза [**88**, **158**].

В тестах stream пропускная способность может зависеть (кроме размерности массивов) от целого ряда дополнительных параметров — числа нитей в OpenMP, применяемой NUMA-конфигурации (параметра NPS), использования режима SMT, включения/отключения турбо-режима процессора, применения 3D V-cache, а также числа и типа используемых DIMM. В большинстве рассматриваемых далее результатов приводятся данные для наиболее широко используемого теста присвоения линейной комбинации двух векторов третьему (stream triad). В случае применения другого варианта теста stream на это указывается явно.

Необходимые базовые данные об этом тесте с зависимостями от вышеупомянутых параметров приводятся в руководстве AMD по настройке EPYC 9004 для HPC [**121**]. Соответствующие результаты там представлены для 2P-серверов с применением 1DPC и двухранговых DIMM (что дает максимальную пропускную способность) DDR5-4800 с моделями EPYC 9654, 9754 и 9684X, с использованием в этих процессорах 8 или 12 CCD. В этом документе представлены также данные с сопоставлениями относительно применения одноранговых DIMM, но эта информация будет рассмотрена ниже отдельно, после явного упоминания о работе с одноранговыми модулями.

Данные оптимальных размеров одномерных массивов для максимизации пропускной способности в stream, указанные в [**121**], приведены в таблице 12. Из этой таблицы видно, что при увеличении числа CCD и соответственно общей емкости кэша L3 пропорционально увеличивается оптимальный размер массивов. Аналогичное имеет место при увеличении емкости кэша L3 при добавлении 3D V-cache (в EPYC 9684X).

Таблица 12. Оптимальные параметры stream для моделей EPYC 9004

Модели	Число CCD	Оптимальная емкость памяти каждого массива
EPYC 9654	8	2.1 GiB
	12	3.2 GiB
EPYC 9754	8	2.1 GiB
EPYC 9684X	8	6.4 GiB
	12	9.6 GiB

Для 96-ядерных EPYC 9654 при использовании разных количеств CCD, разного числа нитей (до 192), разных значений NPS и включениях/отключениях турбо-режимов работа с применением SMT дает понижение пропускной способности за двумя исключениями из всех возможных комбинаций. Включение турбо-режима приводило к росту достигаемой пропускной способности при любых комбинациях других параметров за исключением случаев с применением в тесте 192 нитей (отметим, что максимальные пропускные способности чаще достигались при другом числе нитей, см. ниже).

Для 128-ядерных EPYC 9754 (Bergamo, CCD=8, NPS=4, число нитей до 128) аналогичное поведение пропускной способности при включении SMT или турбо-режима также имеет место при любых комбинациях параметров, за исключением тестов с применением 96 или 128 нитей.

Для 96-ядерных EPYC 9684X с 3D V-cache (CCD=12, NPS=4, число нитей до 192) включение турбо-режима всегда давало увеличение пропускной способности, а включение SMT приводило к уменьшению пропускной способности за исключением случаев работы с 24 нитями.

Из этих данных [**121**] можно сделать вывод, что включение режима SMT обычно способствует понижению пропускной способности, а перевод процессора в турбо-режим (Boost=ON в EPYC) обычно повышает достигаемую пропускную способность.

Наивысшие значения пропускной способности, ожидаемо, для EPYC 9654 были получены при NPS=4. Поэтому мы приведем одну таблицу 13 с разными числами CCD для всех трех моделей процессоров с NPS=4, SMT=OFF, Boost=ON.

Таблица 13. Пропускная способность памяти (МБ/с) 2P-серверов с EPYC 9654, EPYC 9684X или EPYC 9754

Число CCD	Число нитей	Пропускная способность		
		EPYC 9654	EPYC 9684X	EPYC 9754
12	24	739500	717866	
	48	770343	755924	
	96	771153	756739	
	144	774746	755686	
	192	766863	753407	
8	16	651350		755930
	32	771730		782937
	64	784341		787422
	96	779301		780101
	128	765102		774657

Данные из [**121**].

Для EPYC 9654 здесь использовался размер массивов 3.2 GiB, оптимальный для работы с CCD=12. Подробные численные данные для других режимов (параметров), в том числе с указанием числа активных ядер, имеются в [**121**].

Данные о пропускной способности с EPYC 9654 при использовании оптимальных размерностей массивов для каждого числа CCD (2.1 GiB для 8 CCD и 3.2 GiB для 12 CCD) и двухранговых DIMM приведены в таблице 14.

При использовании одноранговых DIMM достигаемая пропускная способность в этом тесте заметно ниже. Так, «абсолютный максимум»,

Таблица 14. Пропускная способность памяти в 2P-сервере с EPYC 9654 при CCD=8 с массивами 2.1 GiB и CCD=12 с массивами 3.2 GiB

Число нитей (8/12 CCD)	8 CCD	12 CCD
16/24	656994	742480
32/48	772818	772308
64/96	787566	771964
96/144	784398	774630
128/192	778488	766348

Данные из [121].

полученный при работе с 8 CCD и 64 нитями с EPYC 9654, составляет 788 ГБ/с для 2P-сервера с двухранговыми DIMM (NPS=4, SMT=OFF, Boost=ON). При работе с одноранговыми DIMM при оптимальных других параметрах пропускная способность меньше, только 726 ГБ/с.

Данные [121] не показывают увеличения достигаемой в этих тестах пропускной способности памяти за счет использования 3D V-cache (возможно и более высокая частота ядер EPYC 9654 в турбо-режиме влияет на нее сильнее), а применение 8, а не 12 CCD часто давало более высокую пропускную способность. И при этом максимальная величина достигается с EPYC 9654 с применением 64 нитей при наличии 96 ядер в каждом из двух процессоров. Если подобная ситуация имеется и для других моделей EPYC 9004, это может способствовать уменьшению масштабирования производительности с ростом числа ядер в разных моделях EPYC 9004 (особенно при работе со связанными памятью приложениями, например, CFD).

В завершение анализа данных [121] следует отметить, что хотя в соответствующих тестах использованы массивы достаточно больших размерностей, в HPC может понадобиться работа и с более крупными массивами, и важно также получить информацию, не будет ли при этом пропускная способность уменьшаться.

Из других данных теста stream для 96-ядерных EPYC 9004 можно указать на [159], где для EPYC 9R14 (грубо говоря, первоначального варианта EPYC 9654) продемонстрирована зависимость пропускной способности от числа задействованных ядер.

В [160] приводятся данные об изменении средней величины пропускной способности между всеми четырьмя вариантами тестов stream (copy, scale,

add и triad) в зависимости от числа применяемых нитей в 2P-сервере на базе 32-ядерных EPYC 9384X по сравнению с 2P-сервером с 32-ядерными Xeon Max 9462 с использованием в нем только HBM-памяти или в ее сочетании с DDR в режиме кэширования HBM-памятью. Тут кэширование HBM-памятью фактически рассматривалось как аналог кэширования с имеющим большую емкость 3D V-cache в EPYC 9484X. Выигрыш в такой величине пропускной способности у EPYC 9384X по сравнению с пропускной способностью с Xeon Max 9462 в режиме кэширования убывает при росте числа нитей от 1 до 64. Выигрыш такой пропускной способности в HBM-режиме без кэширования растет при числе нитей до 64, уже опережая EPYC 9384X при 64 нитях.

Другие отчасти аналогичные данные для тестов stream triad [**161**] относятся к серверам Fujitsu PRIMERGY (1P с 64-ядерным EPYC 9554P и 2P с 64-ядерными EPYC 9534). Там использовались настройки BIOS и выбор NPS= один, два или четыре, и рассматривались разные варианты поддерживаемых DIMM (в т.ч. RDIMM и 3DS RDIMM) с разными величинами скоростей передачи (MT/s) и разной емкостью. Но основные данные об относительной пропускной способности в stream относятся к NPS=1, выключенному SMT и применению 3DS RDIMM (4Rx4 емкостью 128 ГБ).

При этом рассматривается разное число DIMM на сокет. При увеличении их количества до 12 пропускная способность растет, а при 16 модулях и более с применением 2DPC уменьшается на 30–50%. В [**161**] обсуждено также чередование каналов памяти и энергопотребление при выполнении теста.

Другая интересная информация имеется для 1P- и 2P-серверов (Dell PowerEdge R7615 и R7625) с процессорами EPYC 9654P и 9654 в тестах stream из PTS при использовании параметров BIOS по умолчанию [**162**]. Здесь была исследована зависимость достигаемой пропускной способности от конфигураций DIMM и соответственно емкости памяти. Было найдено, что все 4 теста stream (copy, scale, add, triad) показали одинаковые тенденции, и данные приведены для теста triad. Эти данные представлены на рисунке 14. Они показывают, что увеличение пропускной способности при переходе от 1P- к 2P-конфигурации очень близко к двукратному.

Если на основе этих данных предположить удвоение пропускной способности в 2P-сервере по сравнению с 1P-сервером для представленной AMD максимальной пропускной способности 788 ГБ/с для stream triad в 2P-сервере с EPYC 9654 (см. выше), то очень грубой оценкой максимальной

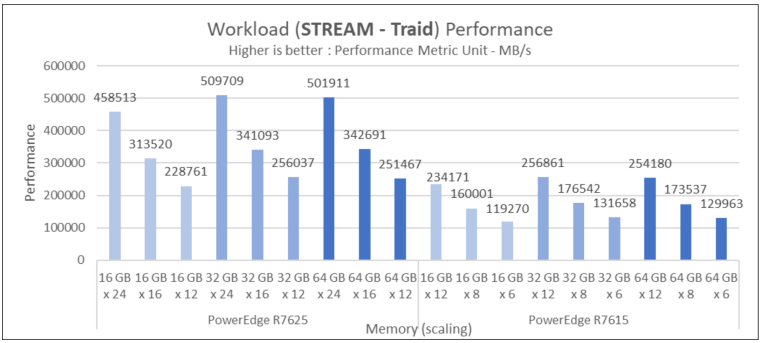


Рисунок 14. Пропускная способность в stream triad (МБ/с) у серверов с EPYC 9654/9654P в зависимости от емкости памяти и параметров DIMM (рисунок из [**162**])

пропускной способности 1P-конфигурации будет 394 ГБ/с, что составляет около 85.5% от пиковой величины для 12 каналов памяти DDR5-4800 (см. таблицу 5).

В [**162**] приведены также аналогичные рисунку 14 данные для зависимостей пропускной способности на Вт и пропускной способности на доллар. Наилучшую пропускную способность и энергоэффективность дает сбалансированная конфигурация с 12 DIMM на сокет с DIMM емкостью 32 ГБ. Этот вариант рекомендуется Dell для связанных памятью рабочих нагрузок. Эта конфигурация обеспечивает пропускную способность на 49 процентов выше, чем с 8 модулями DIMM на сокет с 32 ГБ DIMM. Сбалансированная конфигурация с 12 модулями DIMM на разъем и 16 ГБ DIMM обеспечивает наилучшую пропускную способность за доллар.

Приводившиеся данные, к сожалению, не дают важной информации о зависимости пропускной способности памяти от размера используемых массивов. Выше речь шла о больших емкостях памяти массивов (см. данные их оптимальных размеров в таблице 12). Как показано в [**163**], пропускная способность памяти в 2P-сервере с 56-ядерным Xeon Max 9480 достигала максимума при размере массива порядка 100 МБ, а затем понижалась при увеличении размера, и стабилизировалась при одном ГБ и выше. Но у старших моделей EPYC 9004 емкость кэша старшего уровня гораздо выше, чем у Xeon Max, и влияние кэша на пропускную способность может быть сильнее.

2.4.3. *Тесты умножения плотных матриц (DGEMM) и HPL*

С точки зрения приближения к пиковой производительности тестирование DGEMM для HPC является самым важным. По данным [**148**], 2P-сервер с 96-ядерными EPYC 9654 показал в DGEMM (с использованием aocl) производительность примерно в 1.75 раза больше, чем 2P-сервер с 64-ядерными EPYC 7763 (Milan).

В качестве базовых оценок для EPYC 9004 мы будем использовать данные AMD из [**121**], где для 2P-серверов производительность DGEMM с EPYC 9654 составила 8658 GFLOPS, с EPYC 9684X – 8525 GFLOPS, а с EPYC 9754 – 10730 GFLOPS. Эти данные показывают, что возможности немного более высоких тактовых частот в EPYC 9654 для производительности важнее расширенного 3D V-cache L3 в EPYC 9684X, а применение большего числа ядер в EPYC 9754 важнее меньшей емкости кэша L3 на ядро в этом процессоре.

Данные AMD о производительности в DGEMM для 2P-серверов с другими моделями процессоров представлены в таблице 15.

Таблица 15. Сопоставление производительности 2P-серверов в тестах DGEMM и HPL с разными процессорами EPYC 9004 и Xeon ICL

Модели	EPYC 9214	EPYC 9174	EPYC 9224	EPYC 9354	EPYC 9374F	EPYC 9534	EPYC 9554	EPYC 9654	Xeon 8380
Число ядер	16×2	16×2	24×2	32×2	32×2	64×2	64×2	96×2	40×2
DGEMM (GFLOPS)	1769	2065	2300	3479	3937	5423	6363	7375	
HPL (GFLOPS)	1742	2017	2262	3411.5	3837	5281	6081	7106	4433

Данные из [**152**], для Xeon 8380 – из [**164**].

Приведенные в этой таблице данные не являются максимальными величинами из числа предоставлявшихся производителем. Так, в более позднем документе [**121**] для 2P-сервера с EPYC 9654 приведена производительность 8658 GFLOPS (это уже было указано выше). Кроме того, имеются и еще более высокие показатели производительности в DGEMM и HPL для этих моделей EPYC 9004, полученные в тестах в Китае (см., например, отсылку на Chongqing Microcomputer[8]). Там для EPYC 9654 приводится производительность DGEMM 9283 GFLOPS. Нужно также

[8] https://news.qq.com/rain/a/20230912A0A4M100, accessed 6.03.2025.

отметить данные о производительности DGEMM для процессоров EPYC 9004 в [100], где в рамках тестов HPCC получена и производительность DGEMM на одно ядро. Там для этого использовалась библиотека MKL. Для ядра в EPYC 9654 там получено 51 GFLOPS, в EPYC 9554 – 58 GFLOPS, а в 32-ядерном EPYC 9334 – 60 GFLOPS, что, вероятно, связано с ростом допустимой частоты процессоров с меньшим числом ядер. Но производительность ядер Xeon SPR среднего класса (в 32-ядерных Xeon 6430) выше (71 GFLOPS) [100].

Кроме распространенных тестов DGEMM, в которых принято использовать соответствующий модуль из оптимизированных библиотек, для EPYC 9004 имеются менее интересные данные тестов Phoronix mt-dgemm (см., например, [147]), в которых применяется трансляция простых вложенных циклов для умножения матриц без использования специальных библиотек, что в таком варианте исходного текста вынужденно должно давать более низкую производительность.

Что касается используемого в TOP500 теста HPL, то согласно данным [148] производительность 2P-сервера с EPYC 9654 примерно в 1.77 раза больше 2P-сервера с EPYC 7763. В [88] производительность аналогичного сервера с EPYC 9654 сопоставляется с 2P-сервером с EPYC 7773X, в том числе для разных размерностей (от 140 до 220 тысяч). Для максимальной размерности сервер с EPYC 9654 в HPL быстрее в 1.75 раза. Эти данные были получены с применением aocl 4.0 (при использовании oneAPI MKL 2022.2 производительность была ниже).

Как и рассмотренные выше данные таблицы 15 о производительности 2P-серверов EPYC 9004 в DGEMM, соответствующие данные для HPL – не максимальные из достигавшихся. В более позднем документе, [121], AMD привела более высокие показатели: 8856 GFLOPS для EPYC 9654, 10134 GFLOPS для EPYC 9754, 8620 GFLOPS для EPYC 9684X. Получавшаяся иногда более высокая производительность EPYC 9654 и EPYC 9684X в HPL, чем в DGEMM, вероятно, связана с наличием в ядрах в дополнение к одному блоку FMA еще одного векторного блока сложения (см. выше в разделе 2.1).

Если из данных таблицы 15 поделить приведенную производительность HPL на число ядер, то в расчете на одно ядро для 40-ядерной модели Xeon 8380 получается 55 GFLOPS, для старшей 96-ядерной EPYC 9654 – существенно ниже, 37 GFLOPS. Однако у моделей EPYC с меньшим числом ядер бывает существенно более высокая тактовая частота, и для 64-ядерной EPYC 9554 такая производительность на ядро равна 48 GFLOPS,

а для 32-ядерной EPYC 9374F уже выше, чем для Xeon, 60 GFLOPS (для младшей 16-ядерной модели EPYC 9214 она 54 GFLOPS, почти как для Xeon). В определенном смысле можно говорить о конкурентоспособности по производительности в HPL на ядро у EPYC 9004 и Xeon ICL, поэтому существенно большее число ядер в старших моделях EPYC 9004 дает им соответственно преимущество в производительности.

По данным AMD [**165**], в тесте HPL для 2P-сервера на базе 128-ядерного EPYC 9754 получена производительность 10134 GFLOPS, в то время как для 2P-сервера с 60-ядерными Xeon 8490H полученная Fujitsu производительность равна 7296 GFLOPS [**166**]. Число ядер в сервере с EPYC было в 2.13 раза больше, чем число ядер в сервере с Xeon, а производительность в HPL больше в 1.39 раза. И эта производительность сервера с 60-ядерными Xeon 8490H немножко выше, чем приведенная в [**88**] величина (7257 GFLOPS) для сервера с 96-ядерными EPYC 9654 (представленные в [**152**] производительности, указанные в таблице 15, еще чуть меньше). Но приведенные выше более поздние данные о производительности от AMD [**121**] все равно больше.

В обзоре [**100**] приведены данные о производительности HPL для процессоров: 96-ядерного EPYC 9654 (3811 GFLOPS), 64-ядерного EPYC 9554 (3319 GFLOPS) и 2P-сервера с 32-ядерными EPYC 9334 среднего класса (3527 GFLOPS) с использованием библиотеки MKL, которая давала несколько более высокую производительность, чем OpenBLAS. Более высокая производительность 2P-сервера по сравнению с 64-ядерным процессором, вероятно, также связана с показателями турбо-частот. Этот 2P-сервер также немного опередил 2P-сервер с Xeon SPR среднего класса (Xeon 6430, 3436 GFLOPS), содержащий такое же число ядер, что, вероятно, связано с более высокими тактовыми частотами в EPYC 9334.

2.4.4. *Производительность в тесте HPCG*

Другим тестом производительности, результаты которого для суперкомпьютеров приводятся в TOP500, является HPCG, который многие считают более актуальным для большинства HPC-приложений из-за более низкой вычислительной интенсивности и большей чувствительности к пропускной способности памяти, чем в HPL.

Соответственно в этом тесте можно увидеть и зависимость результата от пропускной способности памяти. В [**121**] для 2P-серверов максимальная производительность, 137.9 GFLOPS была получена при использовании 96-ядерного EPYC 9684X или 128-ядерного EPYC 9754; с EPYC 9654 она была чуть ниже – 135.0 GFLOPS. Эти данные были получены при работе с двухранговыми DIMM, а при использовании более медленных одноранговых DIMM (с процессорами без 3D V-cache) она была на 4% ниже. Эти результаты получены с подсетками размером 192 по каждой оси, с 60-секундным хронометражем.

По данным на 20.11.2024 2P-серверы с EPYC 9654 или EPYC 9754 давали в тесте HPCG 3.1 из PTS при разных исходных данных[9] производительность сильно ниже, чем в [**121**], опережая при этом 2P-серверы с Xeon ICL (Xeon 8380), но отставая от 64-ядерного Xeon SPR (Xeon 8462Y). В этом тесте данные для 2P-серверов с EPYC 9554, 9654, 9754 и 9684X [**12**] продемонстрировали преимущество в производительности относительно GPU Nvidia GH200; Ampere Altra Max M128-30 здесь сильно отстает. Сопоставление производительности EPYC 9004 с Xeon SPR и Xeon EMR проводится далее в разделе 4.1.

2.4.5. *Параллельные тесты NAS (NPB)*

Классическим примером приложений HPC, связанных памятью, являются задачи CFD. Набор тестов NPB базируется на программах, актуальных для CFD, и эти программы также обычно связаны памятью (см., например, [**167**]). Важным преимуществом NPB можно считать то, что для каждого из тестов в NPB, которые имеют двухсимвольное название, имеется набор исходных данных для классов проблем разного размера, именуемых символами от A (маленькая проблема) до F (самая большая). Классы D, E и F относятся к очень большим и требуют[10] соответственно 12.8 ГБ, 250 ГБ и 5 ТБ памяти. Учитывая поддержку до 6 ТБ памяти в EPYC 9004, тесты можно провести даже в 1P-сервере.

[9] https://openbenchmarking.org/test/pts/hpcg, accessed 11.02.2025.

[10] https://www.nas.nasa.gov/software/npb_problem_sizes.html, accessed 21.11.2024.

В отчёте [**100**] также представлены данные о производительности EPYC 9654, 9554 и 9334 в тестах класса C. Производительность отдельных ядер этих процессоров в тестах SP, FT, LU, BT, MG, IS и CG из-за более высоких тактовых частот в более младших моделях с меньшим числом ядер (см. таблицу 6) выше, чем у более старших моделей, и у всех этих процессоров ядра сильно опережают Milan 7713P и Xeon ICL 6330. Аналогичные сравнительные результаты имеют место в этих тестах для 1P и 2P-серверов. Но в тесте EP (с чрезвычайной параллельностью – генерация набора из N псевдослучайных чисел и расчет для них гауссовских отклонений) такого сильного эффекта в производительностях не наблюдается.

На рисунке 15 приведены данные из [**100**] о масштабировании производительности с числом использованных ядер в тесте MG (многосеточный метод, пожалуй наиболее математически приближенный к задачам CFD), а на рисунке 16 – в тесте SP (скалярный пятидиагональный решатель).

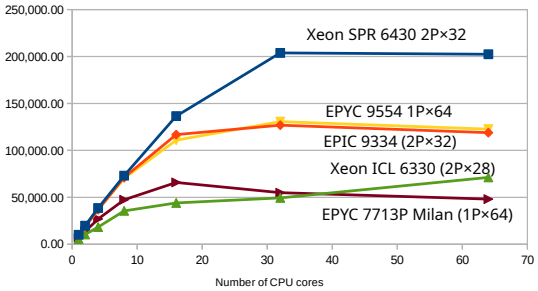


Рисунок 15. Производительность в тесте NPB MG.C (Mop/s) по данным [**100**])

MG.C является связанным памятью (см., например, [**168**]).

Рисунок 15 четко показывают практически прекращение роста производительности в тесте MG класса C при использовании свыше 32 ядер. В тесте SP.C на рисунке 16 мы видим масштабирование производительности до 64 ядер (а для EPYC 9654 – до 96). Эти результаты могут быть важны для выбора оптимального для расчетов в области HPC процессора, поскольку они четко демонстрируют бессмысленность применения в некоторых случаях процессоров с большим числом ядер.

Поскольку выше уже обсуждались данные для HPCG, тест CG из NPB мы здесь далее не рассматриваем, а тест FT рассмотрен далее в подразделе

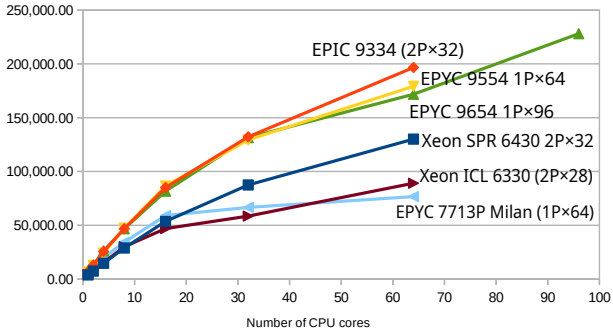


Рисунок 16. Производительность в тесте NAS SP (Mop/s) по данным [**100**])

про БПФ. Из имеющихся на сайте openbenchmarking.org результатов PTS-тестов NPB для EPYC 9004 мы выбрали для иллюстрации результаты тестов MG и LU (см. таблицу 16).

Таблица 16. Данные PTS-тестов NPB-1.4

Модели	Число ядер	Тест MG.C (Mop/s)		Тест LU.C (Mop/s)	
		1P	2P	1P	2P
EPYC 9754	128	127926	247898	281541	598378
EPYC 9654	96	119395	197380	270942	499455
EPYC 9684X	96	136394	233202	342023	537685
EPYC 9554	64	126497	231992	253447	477048
EPYC 9374F	32	122186	214956	179539	372552
EPYC 7763	64	57458	100845	143472	286606

В таблице приводятся средние значения по разным полученным результатам выполнениям тестов [https://openbenchmarking.org/test/pts/npb 11.02.2025] на 24.11.2024.

Эти данные теста MG.C показывают большое увеличение производительности относительно EPYC 7003, масштабирование производительности при переходе от 1P к 2P и повышение производительности за счет использования 3D V-cache в EPYC 9684X. Для EPYC 9554 (1P) результат MG.C в этой таблице достаточно близок к представленному на рисунке 15.

Внешне в этой таблице якобы наблюдается и масштабирование производительности с ростом числа ядер в EPYC 9004. Но эти данные демонстрируют и серьезный недостаток результатов PTS-тестов в openbenchmarking.org, связанный с получением там данных исключительно при использовании всех процессорных ядер в каждом процессоре.

Не так важно, что там бывают большие отклонения в производительности выполнявшихся тестов у разных авторов (для MG.C с EPYC 9654 они были более 10 тысяч Mop/s). Но быстродействие EPYC 9654 в MG.C (даже при использовании максимальной из всех достигнутых в тестах производительности) оказалось ниже 64-ядерного EPYC 9554 (и 32-ядерного EPYC 9374F с ядрами повышенной частоты). Причиной этого могло быть и прекращение роста производительности при числе ядер в процессоре свыше 32, что показывает рисунок 15 – а потом производительность могла начать уменьшаться.

Соответственно данные таблицы 16 показывают относительную производительность разных процессоров в конкретном тесте, не давая оценки эффективности использования тех или иных процессоров. Старшие модели (в первую очередь ориентированные на HPC EPYC 9654) могут показывать гораздо большую производительность относительно моделей с меньшим числом ядер в других тестах (или, наоборот, не демонстрировать этого).

Кроме того, нужно отметить высокие результаты EPYC 9374F в тесте MG.C – они достаточно близки к EPYC 9554 со в два раза большим числом ядер.

В тесте LU.C также видно существенное увеличение производительности при переходе от EPYC 7003, масштабирование производительности при переходе к процессорам с большим числом ядер и от 1P к 2P-конфигурациям серверов. Серверы с EPYC 9554 также опережают по производительности в тестах MG.C и LU.C серверы с Xeon SPR и EMR (см. об их производительности далее в разделе 4.1).

2.4.6. *Тесты производительности преобразования Фурье (БПФ)*

Далее рассматриваются данные для трех разных тестов БПФ – FT из набора тестов NPB, теста FFTW[11] (точнее, варианта FFTW из пакета FFTE, части набора тестов HPCC[12]) и heFFTe (Highly Efficient FFT for Exascale) [**169**].

[11] https://www.fftw.org/, accessed 23.11.2024.
[12] https://hpcchallenge.org/hpcc/, accessed 8.05.2025.

Библиотека heFFTe для трехмерного БПФ исходно ориентирована на экзамасштабирование (и соответствующий американский проект ECP). Поэтому heFFTe предполагает использование гетерогенных узлов с GPU, однако тестировалась и на гомогенных системах, в том числе на суперкомпьютере Fugaku с ARM A64FX [**170**].

Показатели производительности в этом тесте[13] зависят от того, для какого варианта Exascale они представлены (последняя версия на 24.11.2024 -2.4), от версии heFFTe (последняя версия на 24.11.2024 -1.1.0). Там используется тест r2c с разными бэк-эндами (мы рассматриваем FFTW), размерностями N по всем осям (в таблице 17 приведены для N=512) и форматами чисел (в таблице 17 данные heFFTe для FP64). По указанной выше ссылке можно найти данные для разных поколений Xeon и EPYC.

Таблица 17. Производительность в разных тестах БПФ

Модели	heFFTe[1]	NPB FT.C[3]		FFTW из HPCC[4]
	2P	1P	2P	1P
EPYC 9754	208	142074 ± 2376	230807 ± 16627	
EPYC 9654	254	123767 ± 2015	225350 ± 6792	68
EPYC 9684X	323	122140 ± 2543	232995 ± 7382	
EPYC 9554	236	125629 ± 1385	238502 ± 5200	65
EPYC 7773X	125	64292 ± 373	126663 ± 1938	
Xeon Max 9480	2212	–	105720 ± 7286	
Xeon 8490H	129	70543 ± 778	113779 ± 6001	

[1] GFLOPS – для Exascale 2.3 и 2P-конфигураций EPYC 9004 с параметром Power в 400 Вт. Данные из [**171**].
[2] Для работы только с HBM.
[3] Mop/s – для NPB 3.4 в варианте npb.1.4.x из [https://openbenchmarking.org/test/pts/npb от 25.11.2024]. Приведены средние значения и отклонения в разных выполнениях тестов.
[4] GFLOPS, данные из [**100**].

Тест heFFTe здесь демонстрирует большой прогресс от Zen 3 к Zen 4, определенное масштабирование производительности до 96 ядер в процессоре (только в крайне ограниченном смысле, указанном выше в подразделе про тесты NPB). В нем 128-ядерный EPYC 9754 с уменьшенным кэшем L3 показал более низкую производительность, чем 96-ядерные EPYC Genoa, что сочетается и с большим положительным эффектом 3D V-cache в EPYC 9684X, и с сильным увеличением производительности при работе с высокоскоростной памятью HBM в Xeon Max. Последнее соответствует ограничению производительности БПФ пропускной способностью памяти.

[13] https://openbenchmarking.org/test/pts/heffte, accessed 24.11.2024.

Приведенные данные теста NPB FT.C менее информативны, поскольку данные от разных авторов давали большие величины отклонений. Но здесь можно увидеть большой рост в производительности EPYC 9004 по сравнению с 64-ядерным EPYC 7773X (Milan), масштабирование производительности при переходе от 1P к 2P-конфигурациям серверов, и в весьма ограниченном смысле – масштабирование в EPYC 9004 с ростом числа ядер в процессорах от 64 в EPYC 9554 до 128 в EPYC 9754 (несмотря на уменьшение в нем емкости кэша L3). Кроме того, здесь проявляется не такой большой положительный эффект и от емкости кэша L3, и от высокоскоростной памяти HBM (в Xeon Max).

2.4.7. *Производительность в механике сплошных сред*

В этом подразделе в основном речь пойдет о задачах CFD, а аэродинамика сюда также включена из-за близости используемых математических методов, но имеющиеся данные о производительности в ней касаются в основном только предсказания погоды.

Здесь рассматриваются данные о производительности, которые могут быть сопоставлены для разных моделей процессоров, обсуждаемых в обзоре; поэтому они в основном касаются производительности в известных приложениях. В качестве примера исследований в данной области, которые здесь не рассматриваются, приведем работы с применением серверов с EPYC 9654 [**172**, **173**].

Однако мы здесь отметим также относящуюся в некотором смысле и к вычислительной химии работу [**174**] (относится к задачам химического горения), где используются плагины из CFD (в том числе из приложения Nek5000), и исследована производительность на 2P-сервере с EPYC 9654 в сопоставлении с применением GPU Nvidia H100 и AMD MI250X.

Изучение производительности в области CFD представляет интерес для современных многоядерных процессоров в том числе для понимания эффективности применения процессоров с разным числом ядер, поскольку связанные памятью приложения CFD могут прекращать масштабирование производительности при числе ядер, далеком от максимально доступного в настоящее время (см., например, данные по тесту NPB MG.C выше).

Производительность на задачах CFD является одним из самых важных показателей для EPYC 9004 в области HPC, поскольку именно для CFD, возможно, два предыдущих поколения EPYC Zen использовались чаще, чем в других областях HPC. Демонстрация преимуществ производительности EPYC 9654, 9554 и 9374F по сравнению с моделями Zen 3 (EPYC 7763 с 64 ядрами и EPYC 75F3 с 32 ядрами) имеется, например, в [**175**]. Там даже имеющий наименьшую производительность среди применявшихся процессоров EPYC 9004, 32-ядерный EPYC 9374F, опережал обе модели Zen 3.

Сразу после начала поставок серверов с процессорами Zen 4 на известном в мире CFD ориентированном на практическое применение задач гидродинамики сайте появилась общая рекомендация о выборе моделей EPYC 9004 для их применения в этой области [**155**].

Поскольку стоимость лицензии на используемые программные средства CFD часто пропорциональна числу доступных для применения процессорных ядер, отношение стоимость/производительность принимает более высокую практическую значимость. Понятно, что там рекомендуется применять модели процессоров, достигающие большую пропускную способность памяти, соответственно поддерживающие 12 каналов DDR5. Кроме того, важной отмечена и емкость кэша L3, соответственно рекомендация ограничивается только серией Genoa (включая Genoa-X с применением 3D V-cache). Там рекомендуется не старшая 96-ядерная модель EPYC 9654, что в первую очередь связано, возможно, с большим стоимостным показателем.

Данные на рисунках 15 и 16 выше также соответствуют выбору процессоров с не самым большим числом ядер. Такому выбору отчасти соответствуют и приведенные ранее данные для теста stream triad, где наибольшая пропускная способность памяти в 2P-сервере с EPYC 9654 получалась с применением 96, а не 192 нитей при использовании 12 кристаллов CCD, а при работе с 8 CCD (и 64 нитями) пропускная способность была еще немного выше.

Производительность средств OpenFOAM. Мы начнем относящуюся к приложениям часть обсуждения производительности в CFD с данных для программных средств более общего характера, поскольку OpenFOAM используются вообще для решения уравнений в частных производных, и, как отмечено выше в разделе 1, применимы не только для задач механики сплошных сред. Последняя на 2024 год версия OpenFOAM была 12 [**176**].

Для OpenFOAM давно появились разные тесты производительности, а многочисленные данные о ее масштабировании в зависимости от числа процессорных ядер для самых разных процессоров, как x86, так и ARM, постоянно появляются в ориентированном на OpenFOAM блоге использующих CFD[14] с 2018 года по настоящее время.

Используемые ныне и ориентированные на HPC тесты разработаны в 2019 году [**177**]. Они применяются для разных целей, и для них предлагаются сетки разного размера. Полный их список имеется в [**178**]. Данные

[14] https://www.cfd-online.com/Forums/hardware/198378-openfoam-benchmarks-various-hardware.html, accessed 1.12.2024.

этих тестов[15] стали активно появляться для современных процессоров, когда они вошли в состав PTS где результаты теста openfoam-1.2.x для новейших процессоров относятся к OpenFOAM 10. Для этих вариантов теста мы будем ниже рассматривать данные из [171], поскольку это наиболее хорошо подходит для сопоставления разных моделей процессоров.

AMD в [179] приводит средние композитные значения по разным тестам OpenFOAM только для относительной производительности 2P-серверов с EPYC 9004 по сравнению с Xeon SPR. По отношению к 2P-серверу с 56-ядерными Xeon 8480+ сервер с EPYC 9654 быстрее в 1.55 раза (а ядер больше в 1.7 раза), а с EPYC 9684X сервер быстрее в 2.08 раза. У 60-ядерной модели Xeon 8490H соответствующее отношение равно 1.02. Кроме того, по отношению к 2P-серверу с 32-ядерными Xeon 8462Y+ сервер с также 32-ядерными EPYC 9384X быстрее в 1.77 раза. Из этих данных следует также отметить существенное увеличение производительности за счет применения 3D V-cache, что соответствует рекомендациям [155].

В данных [179] может более важно другое – AMD показала сверхлинейное масштабирование производительности для кластера с Infiniband HDR из серверов с EPYC 9684X при числе узлов до 8. Это не является удивительным, поскольку такое имело место еще с процессорами EPYC архитектуры Zen 2 (Roma), в которых также используется иерархия микроархитектуры с применением CCX, и было показано сверхлинейное масштабирование в OpenFOAM [180, 181]. Суперскалярное масштабирование там объясняется более эффективным использованием большой емкости кэша L3 в узлах кластера. При этом, согласно [180], более высокая производительность достигается при использовании только половины ядер в 2P-узлах с 64-ядерными EPYC 7742, что способствует высокопроизводительной работе с кэшем L3.

Здесь полезно также отметить, что сверхлинейное ускорение в кластерах достигается и при умножении разреженных матриц на вектор (SpMV) [182], а такая операция активно применяется в задачах CFD.

Хорошее масштабирование в тесте OpenFOAM MotorBike с разными размерами сетки при числе узлов до 16 показано также в кластере с Infiniband NDR и 32-ядерными EPYC 9354 в 2P-узлах [183].

В качестве числовых показателей производительности 2P-серверов с разными старшими моделями EPYC 9004 для конкретного теста OpenFOAM приведем в таблице 18 данные [171] для теста drivaerFastback при использовании сетки среднего размера.

[15] https://openbenchmarking.org/test/pts/openfoam, accessed 30.11.2024.

Таблица 18. Производительность в тесте OpenFOAM drivaerFastback

Модели	Число ядер	Время расчета, сек
EPYC 9684X	96×2	84
EPYC 9654	96×2	106
EPYC 9754	128×2	115
EPYC 9554	64×2	127
EPYC 7773X	64×2	165
Xeon 8490H	60×2	192

Данные для серверов с EPYC 9004 получены с параметром POWER в 400 Вт.

Из данных этой таблицы видно существенное опережение старшими моделями EPYC 9004 старшей модели Zen 3, которая в свою очередь существенно опережает по производительности старшую модель Xeon SPR. Кроме того, видно положительное влияние на производительность 3D V-cache в EPYC 9684X. Данные, сопоставляющие производительность OpenFOAM на EPYC 9004 и Xeon Max приводится далее в разделе 4.

Производительность в приложениях CFD. По отношению к рассматриваемым далее широко применяемым в мире CFD-приложений от ANSYS при их использования в серверах с EPYC 9004, в том числе в кластерах, есть общие предложения от Supermicro [184]. Там рекомендуется отключать SMT и применять NPS=4. Что касается нецелесообразности применения SMT в разных процессорах для задач CFD, про это известно достаточно давно (см., например, [185]).

Для четырех приложений ANSYS – LS-DYNA, CFX, Mechanical и Fluent – в [186] представлены данные об относительной производительности 2P-серверов с 64-ядерными EPYC 9554 или с 32-ядерными EPYC 9374F по сравнению с 2P-сервером на базе 32-ядерного EPYC 75F3. Для LS-DYNA ускорение составляет соответственно до 1.8 или 1.4 раз, для CFX – до 1.9 или 1.6 раз, для Mechanical – до 1.9 или 1.5 раз, для Fluent – до двух или полутора раз.

Для приложений LS-DYNA, CFX и Fluent в разных тестах количественные, а не относительные показатели производительности представлены в [187] для 2P-серверов с 32-ядерными EPYC 9384X и 96-ядерными EPYC 9684X. Выбор моделей, применяющих 3D V-cache, связан с фактами повышения производительности при его использовании. Применение в тесте 32- и 96-ядерных моделей позволяет оценить производительность для процессоров как старшего, так и среднего класса.

Для оценок производительности в [187] использовался широкий набор тестов: 3-Cars, Neon, ODB 10m для LS-DYNA; LeMans car, Automotive

Pump и Airfoil 10m, 50m и 100m (числа указывают, сколько миллионов ячеек применяется) для CFX; 15 разных тестов для Fluent.

В [**187**] проведено сопоставление производительности этих моделей с моделями Xeon Max 9462 (32 ядра) и старшей 56-ядерной моделью Max 9480, содержащими 64 ГБ высокоростной памяти HBM на процессор. Почти для всех использовавшихся тестов этой емкости для 2P-серверов было достаточно. Исключениями являлись AirFoil 100 для CFX и 3 теста для Fluent. Рост производительности про переходе на работу только с HBM из режима кэширования HBM-памятью в Xeon не превышал 5%. В режиме кэширования среднее композитное ускорение производительности EPYC относительно Xeon для этих трех приложений для 32-ядерных процессоров менялось от 1.3 до 1.7, для старших моделей – от 1.8 до 2.3 раз.

Имеется много представленных AMD данных об ускорении производительности в 2P-серверах с EPYC 9004X относительно Xeon SPR. Для 32-ядерных EPYC 9384X относительно Xeon 8462Y+ : в CFX с Airfoil 10m – до 2 раз, в Fluent с Pump 2m – до 1.8 раз, в LS-DYNA с 3-cars – до 1.9 раз [**165**]. Ускорения в 2P-серверах с EPYC 9684X по сравнению с 2P-серверами с Xeon 8480+ (топ-модели), а также в 2P-серверах с 32-ядерными процессорами EPYC 9384X по сравнению с Xeon 8462Y+ для LS-DYNA представлены в [**188**].

Для CFX аналогичные сравнительные данные для 2P-серверов с EPYC 9684X и 9654 относительно 64-ядерных EPYC 7773X, 56-ядерных Xeon 8480+ и 60-ядерных Xeon 8490H (топ-модели) а также для 32-ядерных EPYC 9374F относительно Xeon 8462Y+ приведены в [**189**]. Для приложения Fluent в [**190**] приведены аналогичные данные для относительной производительности 2P-серверов с 96-ядерными процессорами EPYC 9684X, 9654, 32-ядерными EPYC 9384X, 9374F, EPYC 7773X и с Xeon 8480+, 8490H, 8462Y+. Там также продемонстрировано сверхлинейное масштабирование производительности в ряде тестов Fluent в кластере с Infiniband HDR и 2P-серверами с EPYC 9684X при числе узлов до 8.

Кроме рассматривавшихся выше широко распространенных в мире приложений ANSYS, нельзя также не упомянуть данные о производительности разных процессоров EPYC 9004 в знаменитом программном комплексе Национальной лаборатории Лоуренса в Ливерморе (США), LULESH[16], который направлен на экзамасштабирование. Хотя это приложение узко ориентировано на решение определенной задачи взрыва, там используются численные алгоритмы, достаточно типичные и для других научных приложений для CFD. В LULESH для решения уравнений гидродинамики делается разбиение всей пространственной области.

Соответствующие результаты для 1P- и 2P-серверов с EPYC 9004[17],

[16] https://asc.llnl.gov/codes/proxy-apps/lulesh, accessed 13.12.2024.
[17] https://openbenchmarking.org/test/pts/lulesh, accessed 13.12.2024.

также и [147], показывают плохое масштабирование производительности при переходе от одной модели процессора на другую с большим числом ядер, например, от EPYC 9554 к EPYC 9654, что требует для начала рассмотрения масштабирования с разным числом нитей на процессор. Единственное, имеющиеся там данные показывают более высокую производительность серверов с этими процессорами по сравнению с EPYC Zen 3 и с Xeon ICL, Xeon SPR, Xeon Max и Xeon EMR.

Интересен еще один пример о производительности в области CFD для приложения MFC, предназначенного для решения проблем сжимаемого многофазного потока – то есть не относящегося к таким широко используемым приложениям, как рассмотренные выше приложения ANSYS. В [191] представлены данные о достигаемой в MFC производительности с GPU Nvidia GH200, H100, A100 и V100, а также AMD MI250X, – в том числе по отношению к производительности 1P-серверов с разными процессорами, включая и EPYC 9654, Xeon Max 9468 и Nvidia Grace (ARM Neoverse v2). Среди процессоров самым быстрым оказался EPYC 9654, который отставал от GPU в 1.5–5.3 раза.

В заключение в данном подразделе приведем данные о производительности приложения WRF, предназначенного как для атмосферных исследований, так и для оперативного прогнозирования погоды.

WRF относится к числу связанных памятью приложений. Все приводимые далее данные тестов производительности WRF[18] относятся к применению набора данных Conus 2.5km одной из лабораторий национального центра атмосферных исследований США. Они, в частности, используются в WRF-тесте[19] из PTS, где применяется WRF 4.2.2. Такие тесты проводились, например, как часть PTS-тестов в [147, 171].

По данным AMD [192], в 2P-серверах производительность EPYC 9684X и EPYC 9654 по сравнению с EPYC 7773X выше более, чем в два раза, а по отношению к Xeon 8480+ – в 1.7–1.8 раз. При этом EPYC 9684X с 3D V-cache имеет наивысшую производительность.

Числовые показатели в таблице 19 получены в [147] с использованием детерминизма Power для EPYC 9004. Они демонстрируют высокий рост производительности уже при переходе от 64-ядерных процессоров EPYC 7773X к EPYC 9554. В [171] показана более высокая производительность 2P-сервера с EPYC 9684X (но также в режиме с более высоким энергопотреблением), который опередил и Xeon 8490H.

[18] https://www2.mmm.ucar.edu/wrf/users/benchmark/v44/v4.4_bench_conus2.5km.tar.gz, accessed 4.12.2024.

[19] https://openbenchmarking.org/test/pts/wrf, accessed 4.12.2024.

Таблица 19. Данные теста WRF 4.2.2 (с conus 2.5km) для серверов с EPYC 9004

	EPYC 9654		EPYC 9554		EPYC 7773X	
	1P	2P	1P	2P	1P	2P
Время расчета, сек	7370	3940	8228.5	4503	15436	7411

Данные о производительности WRF 4.2.2 в 1P- и 2P-серверах с 32-ядерными процессорами EPYC 9374F в этом же тесте, в том числе в сравнении с производительностью серверов с 32-ядерными EPYC Zen 3 и Xeon ICL, имеются в [**193**].

По данным [**192**], производительность кластера с Infiniband HDR и 2P-серверами на базе EPYC 9684X и EPYC 9654 до восьми узлов растет почти линейно (с EPYC 9684X чуть быстрее, с EPYC 9654 чуть медленнее). В [**183**] производительность WRF 4.5 в Infiniband NDR-кластере с 2P-серверами, использующими 32-ядерные EPYC 9354, показала практически линейную масштабируемость при числе узлов в кластере до 16.

2.4.8. *Производительность в задачах вычислительной химии*

В этом разделе рассмотрены области, которые можно отнести ко всем разделам химии, включая биохимию и науку о материалах. Здесь рассматриваются данные о производительности в задачах молекулярной динамики, квантовой химии и молекулярного докинга. Но поскольку вычислительная химия активно используется и для задач конструирования лекарств, в конце этого раздела добавлена еще информация о ставших актуальными для медицины задачах геномики, в которых применяются обработка больших данных и, возможно, векторные операции.

Молекулярная динамика. Задачи молекулярной динамики стали одной из самых главных областей, для которой разработчиками процессоров и GPU демонстрируются данные о производительности. Одна из важнейших причин этого – высокий достигаемый уровень распараллеливания. Но с точки зрения применяемых рабочих нагрузок в этой области имеются и кардинальные различия в зависимости от использования классической или квантовой молекулярной динамики, а в последней – еще и в зависимости от применяемого базиса (плоских волн или, что реже, гауссовских функций). Чаще всего тесты производительности выполняются для классической молекулярной динамики, которая и будет далее обсуждаться. Данные для квантовой молекулярной динамики рассмотрены в конце данного подраздела с явным указанием на квантовый уровень расчетов.

AMD для своих процессоров EPYC 9004 приводит специальные обзоры эффективности для задач молекулярной динамики. В [**194**] демонстрируется ускорение в известном высоким уровнем распараллеливания программном комплексе GROMACS на вычислительном экземпляре Amazon AWS EC2 hpc7a.96xlarge с EPYC 9004 по сравнению с EPYC 7003 в hpc6a.48xlarge и хороший уровень масштабирования производительности при использовании до 16 экземпляров виртуальных машин. В [**195**] AMD показала ускорение на 2P-сервере со 128-ядерным EPYC 9754 (Bergamo) по сравнению с аналогичным сервером на базе 56-ядерного Xeon 8480+ более чем в два раза на задачах классической молекулярной динамики (в программных комплексах GROMACS и NAMD), что лишь немного ниже, чем увеличение числа ядер. Для квантовой молекулярной динамики (с методом DFT в базисе плоских волн) в Quantum Espresso ускорение составило 1.4 раза. Можно отметить, что AMD продемонстрировала это при использовании Bergamo с уменьшенной емкостью кэша L3 на ядро, хотя и предлагает такие процессоры в первую очередь для облачных технологий.

В [**121**] AMD привела данные о производительности 2P-сервера с EPYC 9654 в классической молекулярной динамике по программе GROMACS для расчета молекулярной системы (с 1536 тысячами атомов) из молекул воды, показывая зависимость от включения турбо-частот, SMT, применения разных NPS и разного числа ядер в CCX, образующих там общий кэш L3 (см. рисунок 7). Эти данные представлены на рисунке 17.

На оси ординат здесь приводится достигаемая производительность — количество рассчитанных наносекунд (нс) во временной траектории за день. На горизонтальной оси для каждого значения NPS и каждого числа используемых CCD (8 или 12) указывается еще разное число задействованных в CCX ядер (от 1 до 8). Эти данные показывают, что максимальной производительности можно добиться в турбо-режиме с отключенным режимом SMT. Здесь показано, что использовать более тонкие NUMA-настройки (более высокие уровни NPS) не нужно. Зависимость от числа использовавшихся ядер позволяет пользователю определить более оптимальный по их числу процессор.

В [**183**] получены данные о масштабируемости GROMACS в 16-узловом кластере с Infiniband NDR 200 и 2P-серверами с 32-ядерными EPYC 9354 с разными исходными данными (при использовании 5 разных молекулярных систем). Соответствующие результаты представлены на рисунке 18, который показывает хорошее масштабирование во всех случаях. На оси ординат здесь отложена производительность, как и на рисунке 17.

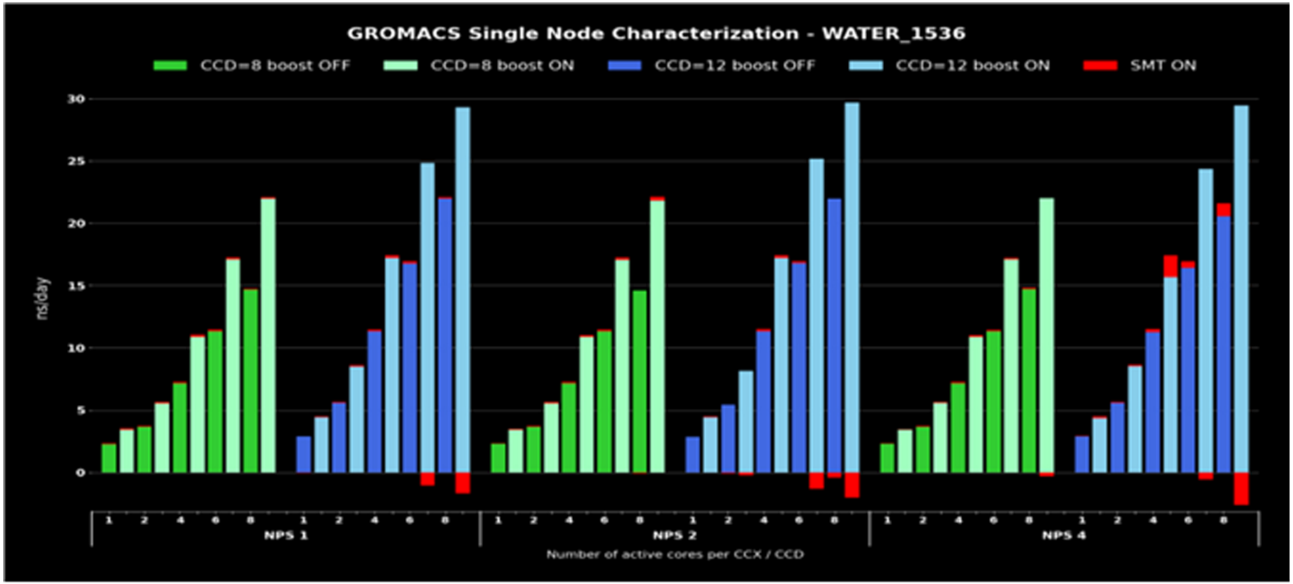
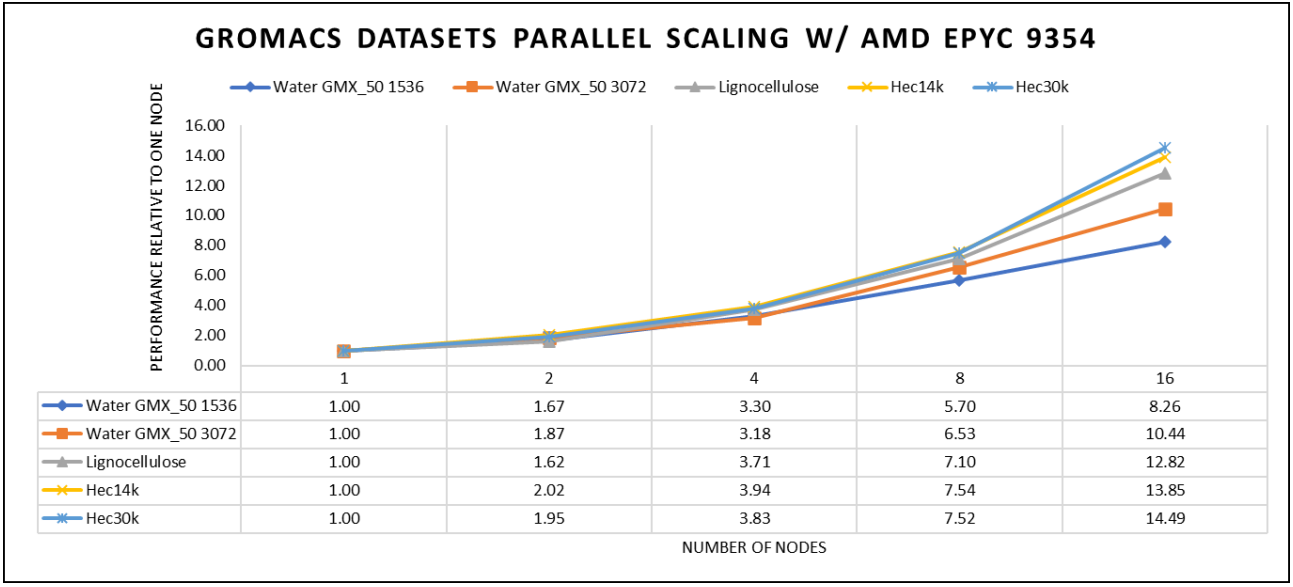


Рисунок 17. Производительность GROMACS в 2P-сервере с EPYC 9654: зависимость от параметров сервера (рисунок из [121])



	1	2	4	8	16
Water GMX_50 1536	1.00	1.67	3.30	5.70	8.26
Water GMX_50 3072	1.00	1.87	3.18	6.53	10.44
Lignocellulose	1.00	1.62	3.71	7.10	12.82
Hec14k	1.00	2.02	3.94	7.54	13.85
Hec30k	1.00	1.95	3.83	7.52	14.49

Рисунок 18. Масштабирование производительности GROMACS в кластере с EPYC 9354 (рисунок из [**183**])

Достигаемое ускорение при распараллеливании зависит, в частности, от размера системы, что четко видно при сравнении распараллеливания в тестах комплексов молекул воды: Water_GMX50_1536 (1536k атомов) и Water_GMX50_3072 (3072k атомов): последний, содержащий свыше 3 миллионов атомов, распараллеливается заметно лучше.

Производительность 1P- и 2P-серверов с EPYC 9684X, 9654 и 9554 в GROMACS с такими комплексами молекул воды представлена в PTS-тестах[20]. Имеющиеся там данные для тестов Water_GMX50_bare в GROMACS 2024 демонстрируют более высокую производительность старших моделей EPYC 9004 по сравнению с Xeon SPR, Xeon Max и Xeon EMR.

Здесь представляется актуальным привести данные о зависимости производительности EPYC 9654 в молекулярной динамике от числа задействованных параллельных процессов в 2P-сервере, которые в [88] имеются для программного комплекса AMBER для расчета в тестах Cellulose NVE с явным учетом растворителя (там молекулярная система содержит около 400 тысяч атомов) и Nucleosome GB с неявным растворителем (там около 25 тысяч атомов). Соответствующие данные из [88] представлены в таблице 20.

Таблица 20. Распараллеливание в AMBER на 2P-сервере с EPYC 9654

Число параллельных процессов	Производительность (нс/день)	
	Cellulose NVE	Nucleosome GB
16	3.31	0.46
32	6.10	0.93
64	10.01	1.85
96	11.99	2.55
128	12.26	3.26
160	11.49	3.75
192	10.58	4.00

Они показывают, что в обоих тестах хорошее масштабирование производительности прекращалось при использовании свыше 64 параллельных процессов, на основании чего рекомендовалось выбирать соответствующее число ядер. Кроме того, данные этой таблицы показывают, что достигаемый уровень распараллеливания здесь зависит не только от размера молекулярной системы.

[20] https://openbenchmarking.org/test/pts/gromacs, accessed 10.12.2024.

В отчете [88] приведены также данные, показывающие более высокую производительность программного комплекса AMBER на этом сервере по сравнению с 2P-сервером с 64-ядерными EPYC 7773X и с 2P-сервером с 24-ядерными Xeon ICL при одинаковом числе параллельных процессов. Однако Xeon ICL в тесте Nucleosome GB при числе процессов до 32 был быстрее обоих серверов с EPYC.

Данные о производительности 1P- и 2P-серверов с 96-ядерными EPYC 9654 и 9684X в другом известном приложении, LAMMPS, имеются в PTS-тестах[21]. Данные о масштабируемости производительности LAMMPS в 16-узловом кластере с Infiniband NDR 200 и 2P-серверами с 32-ядерными EPYC 9354 с разными исходными данными получены в [183].

Последним в обсуждении данных о производительности в классической молекулярной динамике рассмотрим информацию для приложения NAMD. Из результатов PTS-тестов NAMD[22] автором отобраны данные для STMV, которые относятся к молекулярной системе, содержащей свыше миллиона атомов. Эти результаты представлены в таблице 21, где указана средняя среди проводившихся расчетов производительность. Учитывая имеющиеся в этих результатах отклонения от среднего значения, приведенные величины производительности были округлены.

Таблица 21. Производительность серверов с EPYC 9004 в NAMD 3.0b6 в тесте STMV

Модель	Число ядер	Частота, ГГц	Цена	Средняя производительность, нс/день	
				1P	2P
EPYC 9124	16	3–3.7	$1083		1.6
EPYC 9184X	16	3.55–4.2	$4928		1.8
EPYC 9224	24	2.5–3.7	$1825		2.2
EPYC 9374F	32	3.85–4.3	$4850		3.3
EPYC 9534	64	2.45–3.7	$8803		3.9
EPYC 9634	84	2.25–3.7	$10304		5.0
EPYC 9654	96	2.4–3.7	$11805		5.6
EPYC 9684X	96	2.55–3.7	$14756	3.3	6.4
EPYC 9734	112	2.2–3	$9600		5.6
EPYC 9754	128	2.25–3.1	$11900	3.0	6.0

Цены –из [49] на 11.02.2025

Из данных таблицы можно сказать, что достигаемая производительность растет с числом ядер (хотя в конкретной модели кривая зависимости производительности от числа используемых ядер при его увеличении может загибаться). Увеличение емкости кэша L3 за счет применения 3D V-cache вызывает повышение производительности, и этот эффект перевешивает эффект роста числа ядер свыше 96. Понятно, что лучше было бы смотреть на зависимости производительности от числа

[21] https://openbenchmarking.org/test/pts/lammps, accessed 11.12.2024.
[22] https://openbenchmarking.org/test/pts/namd, accessed 4.12.2024.

задействованных в расчете ядер для каждой конкретной модели, поскольку это могло бы продемонстрировать и понижение производительности при росте их числа выше определенного порога – как это было, например, при 64 ядрах в AMBER.

Из этих данных нельзя делать однозначных выводов. Например, применять Bergamo EPYC 9734 вместо Genoa EPYC 9654 даже исключительно для расчетов по NAMD не имеет смысла хотя бы потому, что результаты могут зависеть от рассчитываемой системы, а отклонения в разных расчетах от средней производительности в 2P-сервере с EPYC 9654 составляли аномально высокую величину по сравнению с другими моделями: около 0.4 нс/день.

Кроме того, доступны различные данные об относительной производительности в тестах NAMD в сопоставлении разных моделей EPYC 9004 с Xeon SPR, Xeon EMR и Xeon Max (см., например, данные от AMD [**160**, **195**, **196**]), показывающие преимущества этих моделей EPYC в производительности относительно указанных серий процессоров Xeon.

Информация о производительности разных моделей EPYC 9004 в квантовой молекулярной динамике доступна для приложений CP2K и Quantum Espresso. Имеются данные о производительности для серверов с EPYC 9004 в различных тестах CP2K[23] с использованием DFT в базисе плоских волн (в том числе в варианте с линейным масштабированием) в квантовохимической части.

Данные об относительной производительности Quantum Espresso в 2P-сервере с EPYC 9754 и EPYC 9654 по сравнению с 2P-сервером с Xeon SPR 8480+ фирма AMD привела в [**195**, **196**].

Квантовая химия. Теперь можно перейти собственно к производительности в задачах квантовой химии. В [**88**] исследована производительность в расчетах по приложению VASP методом DFT в базисе плоских волн с применением разных псевдопотенциалов и обменно-корреляционных функционалов для молекулярной системы из тысячи атомов. Расчеты проведены на 2P-серверах с использованием 96-ядерных EPYC 9654, 64-ядерных EPYC 7773X (Milan) и 24-ядерных Xeon ICL. Для создания двоичного кода применялся компилятор Intel и библиотека MKL, без поддержки AVX-512.

При сравнении с одинаковым числом параллельных процессов в разных серверах и одном процессе на ядро, при использовании до 32 ядер (на два процессора) Xeon ICL оказался быстрее (он опережал Milan и при распараллеливании с применением всех 48 ядер). EPYC 9654, естественно, опережал Milan при одинаковом числе задействованных в распараллеливании ядер.

[23] https://openbenchmarking.org/test/pts/cp2k, accessed 12.12.2024.

Но такие расчеты требуют большой пропускной способности памяти, например, из-за применения БПФ. Время расчета с EPYC 9654 обычно уменьшалось при использовании только до 64 параллельных процессов (в одном из расчетов – до 128), и тогда он опережал Xeon ICL.

В [88] исследована также производительность в этих серверах другого широко используемого квантовохимического комплекса программ Gaussian 16. Там расчет проводился для органической молекулы валиномицина (168 атомов) с применением массово применяемого метода DFT (по test0397, одному из прилагаемых к Gaussian) в гауссовском базисе 3-21G. На 48 параллельных процессах время расчета на сервере с EPYC 9654 составило 45 секунд, что в 1.2 раза быстрее, чем с EPYC 7773X, и в 1.6 раза быстрее, чем с Xeon ICL. Но следует сказать, что такие расчеты явно не требуют применения современных мощных процессоров.

При использовании в таком расчете более точного и более распространенного в настоящее время базиса 6-31G (d,p) время расчета с применением 48 параллельных нитей возросло до 164.5 секунд, и уменьшается с увеличением числа используемых нитей до 89 секунд на 192 нитях.

Производительность Gaussian в последнем тесте прилично масштабировалась при числе нитей до 96.Учитывая не самые высокие показатели достигаемого распараллеливания в этом программном комплексе и вычислительную интенсивность квантовохимических расчетов, можно предположить, что такие расчеты в традиционном гауссовском базисе более крупных молекулярных систем и/или более точными методами в серверах с многоядерными EPYC 9004 будут эффективными, что соответствует выводу в [88]. Однако это хотелось бы обосновать соответствующими конкретными расчетами.

Другие данные о временах вычисления на EPYC 9654 в расчете методом TDDFT с использованием численного базиса, центрированного на ядрах атомов (это вычислительно более сложные расчеты, чем DFT в базисе плоских волн), с применением программного комплекса FHI-AIMS для расчета электропроводности теплого плотного водорода имеются в [197].

Для встроенного теста приложения CP2K (H2O-DFT-LS), расчета методом DFT с линейным масштабированием, AMD в [160] привела ускорения, достигаемые в 2P-серверах с содержащими 3D V-cache процессорами EPYC 9004 по сравнению с 2P-серверами с Xeon Max (с HBM-памятью). С 32-ядерными процессорами ускорение EPYC было указано около 1.8 раз (для EPYC 9384X по сравнению с Xeon Max 9462); для старших 96-ядерных EPYC 9684X по сравнению с 56-ядерными Xeon Max 9480 – около 2.1 раза.

В [**183**] продемонстрировано масштабирование производительности квантовохимического приложения CP2K в кластере с Infiniband NDR 200 и 2P-серверами с 32-процессорными EPYC 9354 в узлах. При расчетах систем из молекул воды в рамках встроенных тестов H2O-DFT-LS-NREP и H2O-64-RI-MP2 (методом RI-MP2) достигнуто хорошее ускорение производительности при увеличении числа узлов до 16 (сверхлинейное – в тесте H2O-64-RI-MP2).

В заключение анализа данных о производительности квантовохимических приложений на EPYC 9004 приведем информацию для приложения NWChem, отличающегося самым высоким достигаемым уровнем распараллеливания. Соответствующая информация стала доступна[24] благодаря включению в PTS знаменитого для NWChem теста методом DFT в базисе 6-31(d) молекулы C240 (3600 базисных функций) без учета симметрии и без оптимизации геометрии. Этот тест отличается хорошим распараллеливанием в кластерах и, в частности, на суперкомпьютере Cascade (занимавшем 13-е место в ноябрьском списке TOP500 2013 года) [**198**]. При использовании в узлах по два 16-ядерных процессора Intel Xeon E5-2670 (Intel Xeon Phi, судя по этой публикации, здесь не применялся) при использовании 128 ядер на 4 итерации ССП потребовалось около 500 секунд, а масштабирование производительности было замечено вплоть до применения 2048 ядер.

В PTS-тестах NWChem[25], [**199**, **200**] этим же методом для этой же молекулярной системы было получено плохое масштабирование производительности с увеличением числа использовавшихся ядер, например при «переходе» от 1P к 2P-серверу со старшими моделями EPYC 9004 (и при «переходе» от процессора с 32-ядерными EPYC 9374F, например, к EPYC 9554 и 9654). Причины этого требуют дополнительного исследования.

Молекулярный докинг. Для EPYC 9004 имеются данные о существенном увеличении производительности приложения miniBUDE при включении поддержки AVX-512 [**150**], а в [**201**] демонстрируется гораздо более высокая производительность в miniBUDE для сервера с EPYC 9654 по сравнению с серверами с Xeon ICL и Xeon SPR 8490H (относительно последнего – более чем в два раза). Xeon EMR 8592+ также сильно отстает в производительности от EPYC 9654 [**202**]. Данные о производительности miniBUDE в серверах с разными моделями процессоров архитектур Zen 4 и Zen 3 имеются[26].

[24] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.
[25] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.
[26] https://openbenchmarking.org/test/pts/minibude, accessed 15.12.2025.

Геномика. Безусловно, задачи геномики не имеют отношения к задачам вычислительной химии, а относятся к молекулярной биологии. При использовании вычислительной техники речь фактически идет о биоинформатике или вычислительной биологии. Но геномика выделена в отдельный подраздел здесь потому, что огромная область вычислительной химии актуальна для задач конструирования лекарств, а современная геномика также активно используется в медицине. Серверы и кластеры с топ-моделями EPYC 9004 активно стали применяться в научных исследованиях в геномике (см., например, [**203**–**205**].

Для классического в этой области приложения, GATK [**206**], AMD указывает на увеличение производительности в 2P-сервере с EPYC 9654 на 28% по сравнению с 2P- сервером с Xeon 8490H [**196**].

Созданное позднее приложение Sentieon обладает более высокой производительностью при секвенировании генома [**207**], и в [**208**] показано, что в 2P-сервере с EPYC 9654 полное время выполнения оказалось меньше, чем при секвенировании с приложением Parabricks [**209**] на сервере с GPU Nvidia H100.

2.4.9. *Тесты производительности в задачах ИИ*

AMD, в отличие от Intel, может не так активна в области ИИ. Но аппаратные средства EPYC 9004 имеют необходимые ресурсы (в том числе благодаря поддержке AVX-512), а в программной области можно указать, например, на библиотеку AMD ZenDNN, аналог Intel oneDNN. Естественно, для EPYC 9004 доступны и фреймворки, в том числе TensorFlow и PyTorch, которые объединены AMD как входящие в средства UIF (Unified Inference Frontend) [**210**]. Для настройки EPYC 9004 для задач ИИ AMD подготовила соответствующее руководство [**140**]. И почти сразу после появления этих процессоров стали появляться и научные публикации в области ИИ с использованием серверов на базе EPYC 9004 с анализом данных о достигаемой производительности (см., например, [**211**, **212**]).

По данным [**150**], при включении в EPYC 9654 поддержки AVX-512 производительность в модели ResNet-50 с использованием TensorFlow 2.10 при работе в формате BF16 возрастает на 73%.

AMD указывает на рост производительности (числа изображений в секунду) на 68% в известной модели обнаружения объектов Yolo (You look only once) v5 [**213**], использующей PyTorch, и на 97% (в величине задержки распознавания изображений) при использовании модели ResNet-50 с форматом BF16 в случае подключения в них работы с ZenDNN [**210**]. Это увеличение производительности было достигнуто в 2P-сервере с EPYC 9654.

В [**214**] имеются данные AMD о производительности 2P-сервера с EPYC 9654, в том числе относительно 2P-сервера с 64-ядерными EPYC 7763, с использованием ZenDNN для трех разных моделях ИИ. В ResNet-50 при работе в формате FP32 с данными из использующей иерархию WordNet БД изображений ImageNet (resnet50_fp32_pretrained_model.pb) была получена производительность вывода около 920 изображений в секунду при размере пакета 640 или 927 изображений в секунду при размере пакета 960. Это примерно в 2.1 раза быстрее, чем на сервере с Zen 3.

Для обработки естественного языка в предварительно обученной модели BERT-Large с 340 миллионами параметров для wwm_uncased_L-24_H-1024_A-16 в формате FP32 была получена производительность около 29 выборок в секунду для длины последовательности 256 или около 19 выборок в секунду для длины последовательности 384. Это примерно в 1.8 раз быстрее, чем на сервере с Zen 3. Для модели рекомендаций глубокого обучения с использованием теста DLRM из известного набора тестов MLPerf Inference [**215**] для ИИ (для tb00_40M.pt в формате FP32) также были получены оценки производительности[27], которые не предоставлялись для официальной проверки MLCommons для размещения на сайте [**214**]. Производительность с EPYC 9654 составила около 2948 выборок в секунду для пакета размером 1 и около 3132 выборок в секунду для пакета размером 2, что в 1.7–1.8 раза больше, чем у сервера с Zen 3 [**214**].

Другой известный тест производительности относится к OpenVINO[28], разрабатываемому Intel кроссплатформенному инструментарию с открытым текстом для глубокого обучения (нацеленному на получение высокопроизводительного вывода) [**216**], который поддерживает уже целый ряд известных моделей разных категорий – в том числе больших языковых моделей (LLM, Large Language Models), компьютерного зрения и генеративного ИИ.

В таблице 22 приведены данные о производительности для задач выводов ИИ в PTS-тестах OpenVINO 2022.3, в которых измеряется число кадров (выводов) в секунду (FPS). Там использовались 3 разные модели – обнаружения транспортных средств (Vehicle Detection, VD в таблице), обнаружения велосипедиста (Person Vehicle Bike Detection, PVBD в таблице) и машинного перевода с английского на немецкий (ENtoDE в таблице). Во всех тестах применялись 2P-серверы и формат FP16.

В соответствии с более известными данными для разных GPU, задачи ИИ отличаются высоким уровнем распараллеливания и связанностью

[27] https://mlcommons.org/benchmarks/, accessed 8.05.2024.

[28] https://www.intel.com/content/www/us/en/developer/tools/openvino-toolkit/overview.html, accessed 28.11.2024.

Таблица 22. Данные о производительности (FPS) в тестах OpenVINO

Модель	Число ядер	Модель VD	Модель PVBD	Модель ENtoDE
EPYC 9754	128×2	7097	10290	1249
EPYC 9654	96×2	7970	10005	1001
EPYC 9684X	96×2	8125	9766	1065
EPYC 9554	64×2	6270	7536	805
Xeon Max 9480	56×2	3607	7652	688
Xeon 8490H	60×2	3679	7617	693

Данные из [171], для EPYC получены с параметром POWER в 400 Вт, для Xeon Max – с применением только HBM-памяти.

памятью. Соответственно конкуренция нитей за доступ к памяти может привести к деградации производительности и снижению уровня распараллеливания. Последнее может иметь место не только в модели VD при переходе от EPYC 9654 к EPYC 9754, но это требует дополнительного исследования. В таблице 22 не приводятся данные [**171**] о задержках, поскольку этот показатель мало приемлем с точки зрения его зависимости от числа задействованных в расчете ядер.

Применение 3D V-cache не всегда дает преимущество в производительности. Часто имеющее место сильное опережение моделями EPYC 9004 производительности по сравнению с Xeon SPR и Xeon Max может быть связано с применявшимися программными средствами фреймворков в серверах с Xeon.

Возможные выводы из данных этой таблицы крайне ограничены. Естественно, ее данные говорят только о том, что EPYC 9004 могут часто опережать Xeon SPR по производительности в этих тестах. Есть, конечно, результаты PTS-тестов OpenVINO, где Xeon 8490H опережал EPYC 9004 (см., например, [**217**]).

Другие данные о производительности, где использовались средства OpenVINO, были получены AMD для разработанной Microsoft библиотеки DeepSpeed [**218**] с применением нескольких моделей для LLM, в том числе opt-350m (350 миллионов параметров) и opt-1.3b (1300 миллионов параметров) с размерами пакетов от 1 до 128 [**219**]. Эти данные представлены на рисунке 19, где производительность (в токенах/с) гомогенного 2P-сервера с EPYC 9654 сопоставлена с GPU Nvidia H100. Грубо говоря, H100 опережает гомогенный сервер в два раза. Производительность в этих тестах возрастала с увеличением размера пакета, но на H100 не получилось провести тест opt-1.3b с размером пакета 128 из-за ограниченного размера памяти.

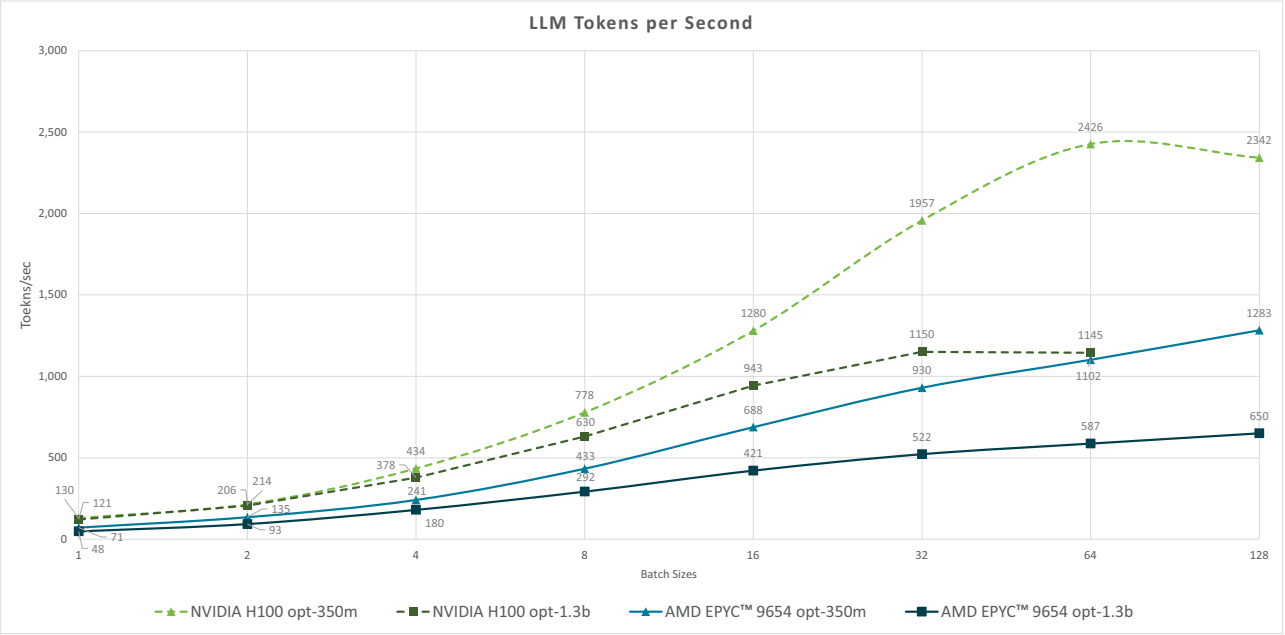


Figure 1: opt-350m and opt-1.3b performance in tokens per second

Рисунок 19. Производительность в токенах в секунду с opt-350m и opt-1.3b (рисунок из [**219**])

Интересным представляется также тест обработки транзакций, в котором различные этапы решения задач ИИ объединяются в единое целое. Такой тест, TPC Express AI (TPCx-AI) [**220**], эмулирует реальные сценарии ИИ и примеры использования науки о данных. Он реализован в виде конвейера, включающего фазы генерации данных, управления данными, обучения, оценки и обслуживания. Здесь после машинного обучения производится и измерение его точности.

В TPCx-AI могут применяться различные наборы данных, что обеспечивает разные уровни масштабирования – SF3, SF10, SF30, SF100, SF300, SF1000 и SF3000, которые отличаются объемами данных. Соответственно тесты до SF30 проводятся обычно на отдельных серверах, а от SF100 и выше – в кластерах. Кроме величины производительности (AIUCpm) определяется отношение стоимости к производительности и энергоэффективность (причем отдельно можно отобрать результаты в Hadoop-кластере).

В TPCx-AI версии 1 на 27.11.2024 все наивысшие показатели производительности до уровня SF1000 были получены Dell с применением 32-ядерных EPYC Zen 4 (в одиночных серверах использовались EPYC 9374F, в кластерах от SF100 до SF1000 – EPYC 9354). Для SF3000 лидерство принадлежало кластеру от Nettrix с 16 узлами[29] (2P-серверами с Xeon ICL 8380). В новом TPCx-AI версии 2 в лидерах на 8.05.2025 для SF10-SF30 оказались уже серверы Dell с EPYC Zen 5.

AMD предоставила ряд документов, указывающих на более высокую производительность EPYC 9004 относительно Xeon EMR в различных задачах ИИ, в том числе при работе с известной библиотекой машинного обучения XGBoost. Но эти данные показывают только относительную производительность и вкратце рассматриваются далее в разделе 4 при сопоставлении производительности с Xeon EMR или SPR.

2.4.10. *О производительности процессоров EPYC 9004 в рабочих нагрузках, характерных для облачных технологий*

Имеется большое количество разных традиционных, не относящихся к HPC или ИИ, областей применения серверов, которые теперь часто используются в рамках облачной технологии, но могут, естественно, применяться и на физических, а не виртуальных серверах.

[29] https://www.tpc.org/tpcx-ai/results/tpcxai_results5.asp?version=1, accessed 27.11.2024.

AMD подготовила специальное руководство по настройке EPYC 9004 при их использовании для задач облачной технологии [**221**]. Поскольку производитель ориентировал свои процессоры Bergamo в первую очередь на облачные технологии, данные о производительности этих процессоров можно найти в ориентированных на соответствующую область применения документах для сообщества AMD [**222**, **223**].

В отличие от рассмотренных в предыдущих разделах рабочих нагрузок, для обсуждаемых здесь скорее характерно отсутствие полной загрузки ресурсов всего процессора в течение всего времени, и применение виртуализации может оказаться эффективным. Например, сюда можно отнести работу с СУБД.

В качестве распространенного теста интегрального характера, где в производительности оценивается собственно виртуализация и набор типовых приложений, можно указать Broadcom/VMware VMmark [**224**]. Он неплохо подходит для широко распространенных ЦОД, использующих облачные технологии, так как в этом тесте виртуализация сочетается с набором плиток для типичных массово используемых приложений. При этом доступные данные содержат различные варианты, в том числе для двух узлов с 1P- и 2P-серверами.

В настоящее время наибольшее число таких доступных данных для рассматриваемых в обзоре процессоров имеется для VMmark 3.1.1 [**225**], где по результатам на конец 2024 года в лидерах производительности однозначно были серверы с EPYC 9004, опережая Xeon SPR и Xeon EMR (серверы с ARM-процессорами в этих тестах не участвуют). Данных для более новой версии VMmark 4 пока еще мало, и они относятся в основном к пятому поколению серверных процессоров, где лидерами в тестах стали Zen 5 [**226**].

Наивысший результат в тесте VMmark 3.1.1 на момент написания обзора показала система из двух 2P-серверов от Supermicro с EPYC 9684X, с производительностью 47.7846 tiles. Аналогичная система из двух 2P-серверов от Dell с 64-ядерными EPYC 7773X показала производительность раза в два ниже, 23.64 24 tiles, что чуть-чуть больше, чем у аналогичной системы с серверами Fujitsu с 60-ядерными Xeon SPR 8490H (23.38 23 tiles), но немного ниже, чем в аналогичной системе с Dell-серверами на базе 64-ядерных Xeon EMR 8592+ (25.34 28 tiles). Здесь приведены максимальные достигнутые в [**225**] результаты для систем с указанными процессорами, но достигаемая производительность, естественно, весьма сильно зависит и от других компонент данных вычислительных систем.

В качестве более узко направленных тестов можно указать данные о производительности EPYC 9004 с использованием разных реляционных СУБД – PostrgeSQL, MariaDB и MySQL. Классикой для таких тестов, ориентированных на обработку транзакций в реальном времени (OLTP) и интерактивную аналитическую обработку (OLAP) являются тесты TPC[30], которые требуют точного выполнения большого количества спецификаций, что может требовать для этого существенных финансовых затрат. Соответственно данные там появляются не так активно, и для систем с новейшими моделями процессоров несколько позднее. Хотя лидерами среди серверов по производительности на 18.12.2024 в тестах TPC-C, TPC-E и TPC-H являются серверы с процессорами EPYC, но в TPC-C с 2022 года это 2P-сервер Supremicro с EPYC Zen 3, а вот в TPC-E в 2024 году лидером стал 2P-сервер Lenovo с EPYC 9554 (в клиентах применялся 24-ядерный Xeon 6442Y), и в TPC-H в 2024 году лидером стал 2P-сервер HPE с 32-ядерными EPYC 9174F. В тесте TPCx-V для виртуализированной серверной платформы с рабочей нагрузкой баз данных также лидирует 2P-сервер с EPYC Bergamo.

Но чаще применяются очень близкие к тестам TPC аналоги. Так, для PostgreSQL есть много результатов специальных тестов PTS, в которых используются pgbench-тесты PostgreSQL (близкие к TPC-B) [**227**], выдающие величины задержек или транзакций в секунду в режимах «только чтение» или «чтение/запись», в том числе для вариантов с разным числом клиентов и разным фактором масштабирования. Для PostgeSQL 17 имеются данные в основном уже для Zen 5 и Xeon 6, а для PostgreSQL 16 или 15 представлено много данных для EPYC 9004.

Однако эти PTS-тесты часто показывают плохое масштабирование производительности при «переходе» к более старшим моделям процессоров с большим числом ядер, и при замене 1P-конфигурации на 2P. В этих тестах часто лидерами оказываются вообще не серверные процессоры x86, и эти результаты здесь не рассматриваются. Есть данные тестов PTS и для MySQL и MariaDB, но соответствующих тематике обзора результатов там гораздо меньше.

Хорошим вариантом для тестирования производительности баз данных является использование свободно доступного приложения с открытым исходным текстом HammerDB[31], в котором разработаны аналогичные TPC-C и TPC-H тесты, TPROC-C и TPROC-H. Они в частности позволяют

[30] https://www.tpc.org/, accessed 17.12.2024.
[31] https://www.hammerdb.com, accessed 18.12.2024.

минимизировать затраты на ввод-вывод, а рабочая нагрузка позволяет в большей степени задействовать возможности процессоров и памяти. HammerDB позволяет работать с основными коммерческими и свободно доступными СУБД, и активно используется в научных исследованиях.

О производительности MariaDB AMD подготовила специальный документ [**228**], в котором показана производительность 1P-сервера с 32-ядерным EPYC 9374F и 2P-сервера с 16-ядерными Xeon SPR 6444Y в тестах TPROC-C и TPROC-H от HammerDB. В TPROC-C сервер с EPYC 9374F в полтора раза быстрее, чем с Xeon 6444Y, а в TPROC-H производительности близки. При этом стоимость EPYC 9374F на конец 2023 года составляла $4850, а двух процессоров Xeon 6444Y – $7244 [**228**]. Но в 2024 году цена процессоров Xeon 6444Y уменьшилась (до $3034 на два процессора), а у EPYC 9374F не поменялась (данные на 17.12.2024 из [**49**]).

Данные проведенных Dell тестов производительности с применением MariaDB и средств mysqlslap из MySQL для 2P- и 1P-серверов с EPYC 9654/9654P, а также1P-сервера с EPYC 9354P, приведены в [**162**]. Там для сервера с EPYC 9354P исследована также зависимость производительности и энергоэффективности от емкости и конфигурации памяти. Кстати, в [**162**] имеются также данные о производительности сервера Apache HTTP на 2P- и 1P-серверах с EPYC 9654/9654P.

Для СУБД MySQL в [**148**, **196**] AMD привела данные о повышенной производительности 2P-сервера с 96-ядерными EPYC 9654 по сравнению с 2P-сервером с 40-ядерными Xeon ICL 8380 в 2.4 раза в TPROC-H и в 2.7 раза в TPROC-C. В [**165**, **223**, **229**] AMD указала на более высокую производительность в TPROC-C с этой СУБД 2P-сервера со 128-ядерными EPYC 9754 свыше двух раз по сравнению с 2P-серверами с ARM-процессорами Ampere Altra Max M128-30 и с 56/60-ядерными Xeon SPR/EMR 8480+/8490H.

Наконец укажем еще данные для другой важной рабочей нагрузки – в области планирования ресурсов предприятия (ERP), производительности в вторых уровневом SAP SD. Для 2P-сервера с EPYC 9654 она составила 148 тысяч SAPS, для 2P-сервера с 64-ядерными EPYC 7763 – 75 тысяч SAPS, а для 2P-сервера с 40-ядерными Xeon ICL 8380 – 48 тысяч SAPS (SAP Application Performance Standard) [**148**].

Выше были рассмотрены данные о производительности в основном для старших моделей EPYC 9004. Информация для средних моделей EPYC 9004 представлена далее при сопоставлении с Xeon SPR в разделе 4.3. Мы не рассматриваем в обзоре данные о производительности EPYC серии 8004,

которые AMD нацеливает на применение для задач интеллектуального края (Intelligent Edge), анализу данных и разработке решений на том месте, где генерируются данные. Некоторые данные о производительности в этой области применения имеются в [**230**].

В заключение всего раздела 2 про серверные процессоры архитектуры Zen 4 следует отметить эффективность (в том числе и для стоимостных показателей) базирующейся на чиплетах иерархии построения микроархитектуры и соответственно конкретных моделей процессоров EPYC.

Приведенные данные показывают существенные преимущества с точки зрения производительности по сравнению с предыдущим поколением Zen 3, но обычно и по отношению к процессорам Xeon SPR и Xeon EMR. Основной сопоставительный анализ производительности EPYC 9004 по сравнению с Xeon SPR, Xeon Max и Xeon EMR проводится далее после анализа их микроархитектуры в отдельном разделе 4. Там приводятся и абсолютные показатели данных «массовых» тестов производительности SPEC и PTS/OpenBenchmarking.

3. Масштабируемые процессоры Intel Xeon четвертого и пятого поколений

В разделе рассматриваются масштабируемые процессоры Xeon четвертого поколения (Xeon SPR) и пятого поколения (Xeon EMR), а также Xeon SPR с памятью HBM (Xeon Max), отнесенные к условному четвертому поколению x86. Общую иллюстрацию об их старших моделях дают таблицы 23 и 24, где приведены также данные о процессорах EPYC Zen 4, Zen 3 и третьем поколении Xeon ICL. Важный, но совпадающий показатель для всех современных серверных x86-процессоров – поддержка режима SMT (именуемого Intel как Hyper-Threading) в этой таблице не приводится.

Большее количество ядер в старших моделях EPYC 9004 явно обусловлено применением более продвинутой технологии TSMC, что в результате часто давало преимущества по важным для производительности показателям. Меньшее число ядер в Xeon соответственно может делать достаточными для эффективной работы другие более низкие показатели, и здесь важна их взаимная согласованность в процессоре, что позволяет максимизировать реально достигаемую производительность.

Таблица 23. Показатели современных серверных процессоров x86 (старшие модели)

	Intel Xeon processors				AMD EPYC processors		
Кодовое имя	Ice Lake-SP	Sapphire Rapids-SP		Emerald Rapids-SP	Milan	Bergamo	Genoa
Семейство процессоров или архитектура	Третье поколение масштабируемых Xeon	Четвертое поколение масштабируемых Xeon	Xeon Max (Xeon SPR с HBM)	Пятое поколение масштабируемых Xeon	Zen 3	Zen 4	Zen 4
Линейка моделей	Xeon 8300	Xeon 8400	Xeon 9400	Xeon 8500	EPYC 7003	EPYC 9004	
Старшая (топ) модель	Xeon 8380	Xeon 8490H	Xeon Max 9480	Xeon 8592+	EPYC 7763	EPYC 9754	EPYC 9654
Число ядер	40	60	56	64		128	96
Базовая частота, ГГц	2.3	1.9			2.45		2.4
Турбо-частота, ГГц	3.4	3.5		3.9	3.5	3.1	3.7
Турбо-частота всех ядер, ГГц	нет	2.9	2.6	2.9	нет	3.1	3.35
FLOPS/такт на ядро[1]	32				16	24	24
Пиковая производительность, GFLOPS[2]	2944	3648	3405	3891	2509	6912	5530
Технология, нм	10				7	5	
Цена, $	9359[3]	17000	12980	11600	7890	11900	11805

[1] Для FP64;
[2] Рассчитана для базовой частоты ядер;
[3] цена на 4.08.2024 из [231].

Таблица 24. Показатели современных серверных процессоров x86 (продолжение)

	Intel Xeon processors				AMD EPYC processors		
Кодовое имя	Ice Lake-SP	Sapphire Rapids-SP		Emerald Rapids-SP	Milan	Bergamo	Genoa
TDP, Вт	270	350			280	360[2]	
Межсоединение процессоров сервера	UPI: 3×20	UPI:4×24		UPI:4×24	IF:4		
Максимальная ПС[1]	11.2 GT/s 67.2 ГБ/с[3]	16 GT/s 192 ГБ/с	16 GT/s 192 ГБ/с	20 GT/s 240 ГБ/с	128 ГБ/с[1]	256 ГБ/с	
Кэш L1 (на ядро)	64 КБ	80 КБ			64 КБ		
Кэш L2 (на ядро)	1 МБ	2 МБ			512 КБ	1 МБ	
Кэш L3	60 МБ	112.5 МБ	112.5 МБ	320 МБ	256 МБ	256 МБ	384 МБ (32 МБ/CCD)
L3 с 3D V-cache	Нет				768 МБ	До 1152 МБ	
Тип памяти DDR	DDR4-3200×8	DDR5-4800×8		DDR5-5600×8	DDR4-3200×8	DDR5-4800×12	DDR5-4800×12 до 6ТБ
ПС DDR	205 ГБ/с	307 ГБ/с	307 ГБ/с		205 ГБ/с	461 ГБ/с	
HBM-память	нет		64 ГБ	нет			
Пиковая ПС HBM	—		1638 ГБ/с	—			
PCIe: версия, число линий	4.0, 64	5.0, 80			4.0, 128	5.0, 128	

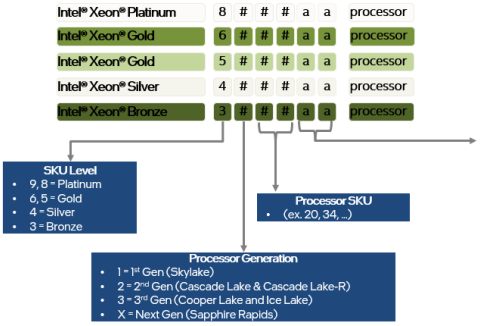
Основные данные и цены на 27.04.2025 из БД процессоров [49].
[1] ПС – пропускная способность на один процессор;
[2] Можно конфигурировать от 320 до 400 Вт;
[3] Данные из [232];
[4] По данным [23].

Одним из самых важных для задач HPC усовершенствований при выпуске масштабируемых процессоров Xeon можно считать появление расширения ISA AVX-512, включающего FMA-операции над 512-битными векторами. Об AVX-512 уже говорилось немного выше, в разделе 2.1 про ядра Zen 4, хотя там было реализовано только подмножество AVX-512 (подробнее об AVX-512 см., например, [**110**, **233**]). Безусловно, это сильно способствовало достижению на многие годы более высокой производительности ядер Xeon в HPC по сравнению с ядрами серверных процессоров AMD. Дальнейшее обсуждение в этом подразделе базируется на данных [**233**].

В различных поколениях масштабируемых процессоров Xeon было заложено много актуальных усовершенствований, ставших общими для последующих поколений, использующимися и в рассматриваемых в обзоре процессорах. С первого поколения масштабируемых процессоров Xeon (Skylake-SP) предполагалась возможность увеличения числа процессорных ядер в будущем, что потребовало перейти на другой вариант их межсоединения, в качестве топологии которого стала применяться сетка (mesh). Такая сетка представлена на рисунке 22 ниже. С самого начала была сделана также ориентация на возможность построения серверов с числом сокетов от 2 до 8.

В соответствии с возможностью производства очень большого количества разных моделей Xeon с разным числом процессорных ядер и разным допустимым числом процессоров на сервер масштабируемые Xeon получили дополнительные брендовые уточнения наименований (в порядке увеличения их вычислительных возможностей, производительности и цены): Bronze, Silver, Gold и Platinum. Первые два обеспечивали возможность работы в не более чем в 2P-варианте серверов, а Gold и Platinum – еще и в серверах с 4 сокетами. И лишь Platinum могли применяться в 8-процессорных серверах [**233**]. Но номер конкретной модели Xeon однозначно приписывает ее к определенному варианту брендов. Это можно увидеть на рисунке 20.

Общая система нумерации моделей в виде четырех десятичных цифр LGmn, где L означает уровень процессора (Platinum, Gold, Silver, Bronze), G – номер поколения процессоров (mn нумерует конкретные SKU) расширена добавлением целого ряда суффиксов (см. рисунок 20). Возможность применения в разных SKU разных акселераторов и их сочетаний добавляет уточнений для конкретизации SKU. Это нацелено на достижение эффективности использования разных SKU в разных узких областях применения. L=8 или 9 означает Platinum, L=6 – Gold [**235**]. G=4 отвечает четвертому поколению масштабируемых процессоров Xeon (Xeon SPR), G =5 – 5-му поколению (Xeon EMR).



Workload Optimized		Built-in Accelerators*				Up to Socket/UPI	SKU Shelf
		AMX	DLB	DSA	IAA	QAT	
P	Cloud – IaaS	✓	1	1	1	1	2S / 3 UPI	P
V	Cloud – SaaS	✓	1	1	1	1	2S / 3 UPI, 1S	P
				1	1		2S / 3 UPI	G6
M	Media Transcode	✓		1	1			G6
H	DB and Analytics	✓	4	4	4	4	8S / 4 UPI	P
			2	1	1	2	4S / 3 UPI	G6
N	Network/5G/Edge (High TPT / Low Latency)	✓	4	4	0	4	2S	P
			2	1	0	2	2S, 1S	G6
								G5
S	Storage & HCI	✓	4	4	0	4	2S / 4 UPI	G6
			2	1	0	2	2S / 3 UPI	G5
T	Long-life Use/High Tcase	✓		1			2S / 2 UPI	S
U	1-Socket*	✓		1			1S	G6
Q	Liquid cooling	✓		1			2S / 3 UPI	P
								G6
+	Feature +	✓	1	1	1	1		

Рисунок 20. Система нумерации Xeon SPR и Xeon EMR (рисунок из [234])

На суперкомпьютерах и в кластерах для HPC чаще используются процессоры с наивысшим уровнем, Platinum (они в основном и рассматриваются далее в данных о производительности), хотя широко применяются и Xeon Gold. Суффиксы, отвечающие ориентации именно на задачи HPC, среди данных процессоров, можно сказать, отсутствуют.

Хотя в списке суффиксов имеется M, согласно [166] ориентированный на средства массовой информации, ИИ и HPC, в [235] HPC не упоминаются. Данные, подтверждающие активность применения таких процессоров для ИИ или HPC, автору неизвестны. Чаще можно было видеть использование в этих областях и соответствующих тестах производительности суффиксы

- *H* – для работы с БД или суффиксы для ориентации на облачные технологии,
- *Y* – для IaaS,
- *P* – для IaaS с повышенными частотами,
- *V* – для SaaS.

Можно обратить внимание и на процессоры с суффиксом Q, ориентированные на жидкостное охлаждение (и соответственно более высокие частоты), что возможно актуально для некоторых высокопроизводительных серверов. Однако нужно смотреть на реальную производительность моделей (см. раздел 4.1 далее).

Символ + в конце номера модели означает, что она имеет по одному включенному акселератору DSA, DLB, QAT и IAA (см. о них далее в разделе 3.1.3). Полное описание суффиксов имеется в [235].

Но вернемся к общим характеристикам микроархитектуры этих поколений процессоров. Каждое ядро и соответствующий фрагмент (slice) кэша L3 (Intel предпочитает использовать термин кэш последнего уровня,

LLC) имеют специальный блок CHA (Caching and Home Agent). Он сопоставляет адреса доступа с определенным банком LLC, контроллером памяти или подсистемой ввода-вывода, и дает информацию о необходимой маршрутизации для достижения пункта назначения с помощью межсоединения сетки. «Распределенное» построение всего CHA в процессоре позволяет обеспечить масштабирование с ростом числа используемых ядер.

Такой механизм позволяет уменьшить задержки обмена данными между ядрами и кэшем L3, памятью и системой ввода-вывода по сравнению с необходимостью обхода кольцевого межсоединения и, возможно, коммутатора между двумя кольцами в используемой ранее топологии межсоединения ядер процессоров Xeon [**233**].

В качестве нового межсоединения для связи между масштабируемыми процессорами Xeon в сервере применяется UPI (Ultra Path Interconnect). Для подключения к другим процессорам может использоваться по нескольку двунаправленных каналов UPI. В UPI применяется протокол когерентности home snoop на основе каталогов (в [**233**] представлена об этом достаточно подробная информация). Для примера на рисунке 21 иллюстрируется топология межсоединения UPI для 8-сокетного сервера.

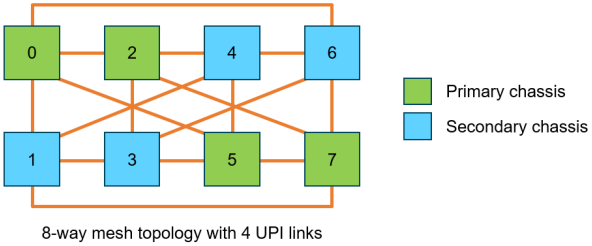


Рисунок 21. Топология межсоединения в 8-процессорном сервере Lenovo с Xeon SPR (рисунок из [**236**])

Эта топология относится к процессорам Xeon SPR, имеющих по четыре канала UPI, а в масштабируемых процессорах Xeon с меньшим числом UPI на процессор (например, в Xeon Skylake) топология другая. В Xeon EMR возможность построения серверов, имеющих больше двух сокетов, не поддерживается.

Другая актуальная новая возможность масштабируемых процессоров Xeon – кластеры SNC (sub-NUMA clustering). SNC создает несколько доменов локализации в процессоре (здесь в соответствии с [**233**] их предполагается два), отображая адреса одного из локальных контроллеров памяти в одной половине фрагментов LLC ближе к этому контроллеру

памяти. Адреса, отображенные на другом контроллере памяти, привязаны к фрагментам LLC в другой половине процессора. Благодаря этому процессы, работающие на ядрах в одном из доменов SNC, используют память из контроллера памяти в том же домене SNC и получают более низкую задержку LLC и памяти по сравнению с задержкой при доступе за пределы этого домена SNC. Для работы в режиме SNC память должна быть заполнена симметрично.

SNC имеет уникальное местоположение для каждого адреса в LLC. Локализация адресов в LLC для каждого домена SNC применяется только к адресам, отображенным на контроллеры памяти в том же сокете.

Такая конфигурация с двумя доменами SNC0 и SNC1 представлена на рисунке 22.

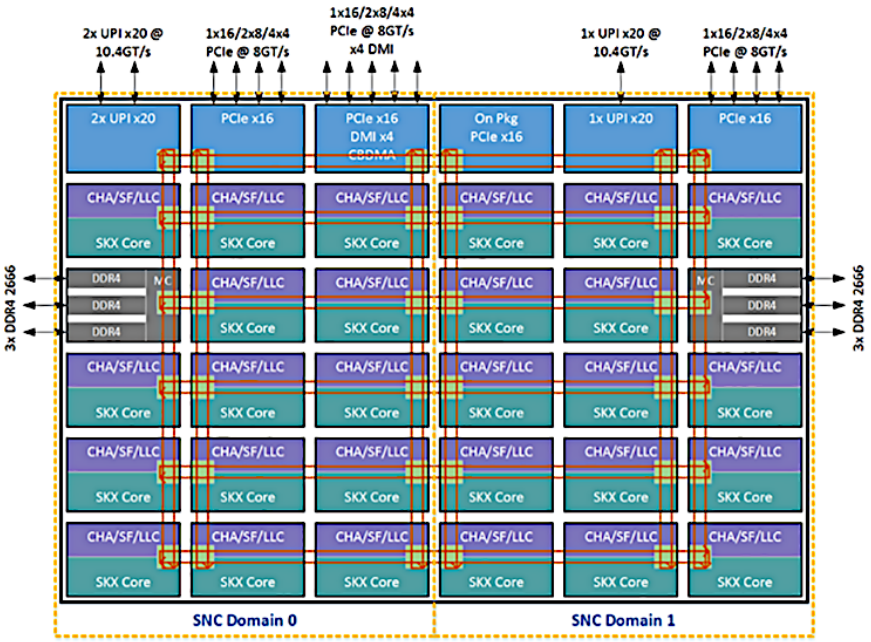


Рисунок 22. Кластеры SNC в масштабируемых процессорах Xeon (рисунок из [233])

Кэш LLC в масштабируемых процессорах Xeon вообще стал неинклюзивным (ранее он был инклюзивным). Детальнее об изменении иерархии кэша см. [233].

Среди общих аппаратных средств, используемых несколькими поколениями масштабируемых процессоров Xeon, в частности Xeon SPR, Xeon

Max и Xeon EMR, здесь полезно указать и на средства виртуализации, в том числе масштабируемую виртуализацию ввода-вывода [**237**]. Рассмотрение этих средств требует подробного анализа и здесь не проводится. В [**233**] приведено еще много новых возможностей масштабируемых процессоров Xeon, которые здесь даже не упоминаются. Мы ограничились здесь только теми общими вещами, которые будут рассматриваться далее для конкретных относящихся к обзору поколений Xeon.

3.1. Масштабируемые процессоры Xeon 4-го поколения (Xeon SPR)

Если Xeon ICL был монолитным процессором, то в Xeon SPR применяются чиплеты. Этот процессор содержит 4 плитки, объединенные с помощью технологии 2.5D -Embedded Multi-die Interconnect Bridges (EMIB) [**238**]. Что содержит каждая плитка, описано далее в разделе 3.1.3.

Основные научные публикации сотрудников Intel, касающиеся микроархитектуры Xeon SPR – это работы [**238**] и [**239**]. В [**238**] рассматривается в основном более низкий уровень, приближенный к полупроводниковой технологии. Мы приводим в этом разделе только более близкую к микроархитектурному уровню информацию. Отметим здесь только, что в Xeon SPR применяются специальные методы более высокого, чем просто полупроводникового уровня, позволяющие устранять дефекты и повысить выход годных к работе процессоров [**238**].

В [**239**] имеется много интересной информации для анализа микроархитектуры. Но учитывая очень быстрые усовершенствования, которые Intel делала в своих процессорах, далее анализ в обзоре базировался в большей степени на более новых и подробных технических данных с сайта Intel.

Максимальное число ядер в Xeon SPR по сравнению с Xeon ICL и емкость кэша L2 возросли в полтора раза; существенно увеличилась и емкость кэша L3 (см. таблицу 25 далее).

Выше отмечено, что Intel демонстрирует ориентацию своих моделей Xeon их суффиксом. Например, с суффиксом N предлагаются оптимизированные для работы с сетями Xeon SPR [**240**]. Они поставляются в конфигурациях со средним числом ядер MCC (Medium Core Count) и экстремальным количеством ядер XCC (eXtreme Core Count). Процессоры MCC имеют 24–32 ядра на сокет при мощности 165–205 Вт, а процессоры XCC имеют 52 ядра на сокет при мощности 300 Вт.

Яркой особенностью Xeon SPR является то, что они снабжаются еще специальными разработанными Intel акселераторами, и фирма предлагает с возможным применением разных акселераторов разные варианты процессоров, направленные на разные области применения.

Акселераторы нацелены на повышение производительности в определенных областях применения, что способствует формированию многочисленных разных моделей процессоров, оптимизированных на четкие и часто достаточно узкие направления работы. Некоторые из акселераторов, которые могут использоваться процессорами Xeon SPR, являются заменой или модернизацией применявшихся Intel ранее. При этом в разных моделях Xeon может быть не один, а до четырех акселераторов, или они все выключены.

Для работы с акселераторами необходимо использовать специальное (в дополнение к драйверам) программное обеспечение Intel, часто реализованное в виде библиотек. Очевидным недостатком применения акселераторов является необходимость модернизации кода приложений и появление проблем переносимости такого кода на другие платформы. В обзоре в соответствии с его общей ориентацией акселераторы подробно не рассматриваются, но краткая информация о них приводится далее вместе с единичными ссылками на первые появившиеся работы, демонстрирующие достигаемую при их применении производительность.

Еще одной особенностью Xeon SPR является поддержка работы с постоянной памятью Intel Optane. В частности, ориентация на возможную работу с Optane имеется и в акселераторах Intel In-Memory Analytics Accelerator (IAA). Однако в связи с экономической неэффективностью производство Optane было прекращено в 2023 году[32]. Эти аппаратные средства в обзоре не рассматриваются.

3.1.1. *Микроархитектура ядер в Xeon SPR*

Используемая в Xeon SPR микроархитектура ядер – это Golden Cove, она преемница микроархитектуры Sunny Cove в Ice Lake [**241**], и в целом является общей с ядрами Xeon Max.

Блок-схема, иллюстрирующая микроархитектуру Golden Cove, приведена в [**239**]. Ниже на рисунке 23 представлена очень близкая по содержанию блок-схема, функциональность большинства блоков которой понятна из их названия. Отметим только, что P0, ..., P11 представляют собой пронумерованные порты.

Весь анализ микроархитектуры Golden Cove здесь основан на [**241**], и улучшения имеются в виду относительно микроархитектуры ядер Ice Lake-SP (Sunny Cove). В основном эти улучшения проявляются количественными показателями, например увеличением «ширины» выполнения (числа портов в различных блоках) или емкости блоков.

[32] https://www.techpowerup.com/310819/intel-optane-still-not-dead-orders-expanded-by-another-quarter, accessed 5.08.2024.

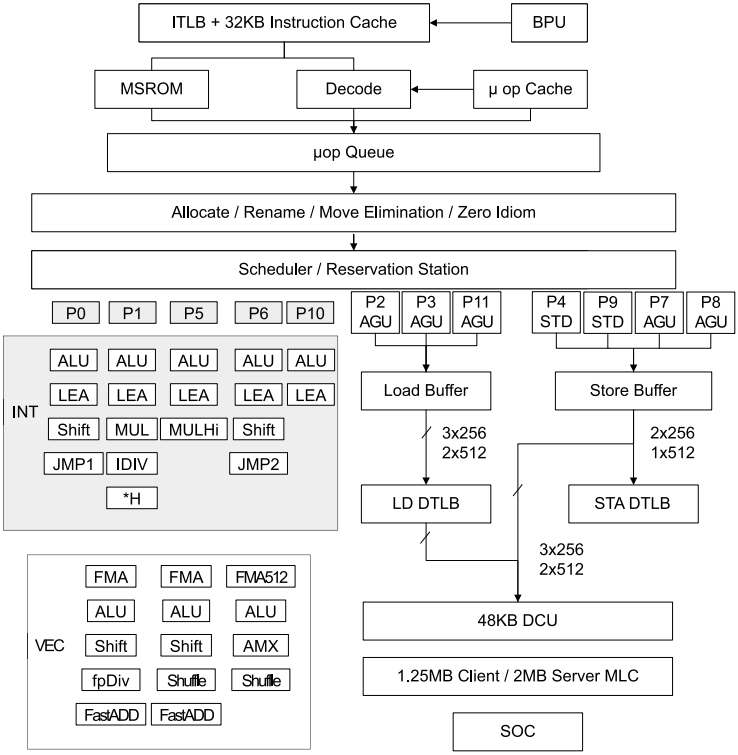


Рисунок 23. Общая функциональность конвейеров ядер с микроархитектурой Golden Cove (рисунок из [241])

В очень важной для производительности современных процессоров с поддержкой OoO фронт-энд части Golden Cove пропускная способность конвейера декодирования возросла в два раза, до 32 байт/такт, и количество декодеров возросло с 4 до 6, позволяя выдавать 6 декодированных команд за такт. Очередь декодирования команд IDQ (Instruction Decode Queue) в Golden Cove может сохранять 144 микрооперации на логический процессор в однонитевом режиме (или 72 микрооперации на нить при включенном режиме SMT). Размер кэша микроопераций увеличился, а его пропускная способность возросла до восьми микроопераций за такт.

Было улучшено и предсказание переходов, особенно для рабочих нагрузок с большим объемом кода. Емкость ITLB была удвоена – до 256 записей для страниц размером в 4 КБ, и до 32 записей для страниц размером 2/4 МБ. Была улучшена обработка невыполненных промахов

страниц (ситуаций «промах при промахе»). Максимальная пропускная способность загрузки увеличена с 2 до 3 загрузок/такт.

Емкость кэша L2 в ядре была увеличена до 2 МБ. В Golden Cove было также увеличено число портов для исполнительных блоков с 10 до 12 (см. рисунок 23), а у некоторых из этих портов были расширены функциональные возможности. В микроархитектуре ядра имеются и другие усовершенствования, например, в завершении обработки микрокоманд (retirement) – см. более подробно в [**241**].

Можно проводить прямое сопоставление подобных количественных показателей микроархитектур Golden Cove и Zen 4, воспользовавшись, например, данными таблицы 4 и имеющимися в других источниках, ссылки на которые приведены в разделе 2.1 выше. По некоторым из этих показателей Golden Cove лучше. Но, как отмечено выше, производители часто оценивают эффективность своих микроархитектур интегрально, достигаемым значением IPC. Поскольку микроархитектура ядер часто используется не только в процессорах для серверов, но и для ПК, там эта величина, с учетом большего количества применяемых последовательных приложений, более актуальна.

IPC зависит от используемого для его вычисления приложения или теста, и сопоставления неких средних оценок ограничены по сути, хотя фирмы часто ссылаются на прогресс IPC в своих ядрах. Обычно имеется в виду IPC при выполнении тестов SPECcpu, хотя результаты для SPECint и SPECfp будут отличаться (см., например, [**242**]). Отличие достигаемых IPC между Golden Cove, Raptor Cove (микроархитектуры ядер Xeon EMR) и Zen 4 составляет не более нескольких процентов. Поэтому для серверов с рассматриваемыми в обзоре процессорами сравнительный анализ IPC ядер не является таким актуальным, и прямое сопоставление отдельных блоков их микроархитектур здесь не проводится.

Но мы укажем на подробное сопоставление микроархитектур Nvidia Grace (Neoverse V2), Intel Golden Cove и AMD Genoa, проведенное в [**13**]. Там приводятся некоторые потенциальные преимущества микроархитекутры Grace, но актуальнее сравнение данных достигаемой производительности. Там отмечено, что Genoa достигает 78% от своего теоретического пика пропускной способности памяти, тогда как суперчип с Grace и Xeon SPR достигают 87% и 90% соответственно. Но согласно данным AMD в [**121**], в 2P-сервере с EPYC 9654 достигается максимум, составляющий более 85% от величины пиковой пропускной способности (см. выше в разделе 2.4.2), что весьма близко к показателю для Grace.

В [**13**] сверялись и конкретные модели: 72-ядерного Grace с 52-ядерным Xeon 8470 и 96-ядерным EPYC 9654. Там также приводятся интересные данные о реальном снижении тактовых частот (с применением

турбо-режима в x86) при высоких вычислительных нагрузках, в том числе с использованием средств AVX-512 при различном числе ядер. При этом частота ядер Xeon 8470 снижается гораздо сильнее, чем у EPYC 9654 как с применением AVX-512, так и без, а у Grace частота почти неизменна.

Здесь нужно также отметить, что ядра Xeon SPR обеспечивают повышенную производительность благодаря программируемому контроллеру управления питанием [**238**].

Процессор Grace вообще не поддерживает 512-битную векторную арифметику. В [**13**] приводятся расчетные данные пиковой производительности для максимально возможных частот и полученных при их снижении при рабочей нагрузке на максимальном числе ядер. Но и при таком расчете Grace сильно отстает по пиковой производительности FP64 от EPYC 9654. Это сочетается с более низким TDP у Grace и более высокой поддерживаемой пропускной способностью используемой для работы с Grace более дорогой памяти LPDDR5X, что и соответствует ориентации на применение Grace только в серверах с GPU Nvidia. Без реальных данных о достигаемой производительности приложений дальнейшее сопоставление становится не актуальным.

В серверных процессорах кроме IPC имеется большой ряд других важнейших параметров – кроме допустимых тактовых частот и числа ядер это, например, энергопотребление и стоимости, зависящие и от используемой технологии. Но самыми важными являются данные о производительности (для отдельных ядер в последовательно исполняемых тестах и приложениях) – это рассматривается далее в разделе 4.1.

3.1.2. *Расширения ISA в ядрах Xeon SPR*

Естественно, на сайте Intel имеется много источников соответствующей информации. С учетом ориентации на усовершенствования в аппаратной реализации Xeon SPR, краткая информация о расширении ISA имеется в [**243**].

Полное руководство по современной ISA имеется в [**244**] (см. также [**241**]). Наиболее важным расширением системы команд, с нашей точки зрения, является AMX. Оно ориентировано на задачи ИИ (на глубокое обучение и выводы моделей ИИ) и используется, начиная с Xeon SPR, в частности, в Xeon EMR и в P-ядрах Xeon 6. С нашей точки зрения, AMX очень важно и имеет большие перспективы на дальнейшее развитие. AMX можно считать некой новой парадигмой программирования в x86.

AMX не просто расширение системы команд; для его реализации имеется специальный аппаратный блок. AMX можно отнести к акселераторам, но также, как и AVX-512 (в отличие от других упоминавшихся выше

ускорителей), блок AMX имеется в каждом ядре Xeon SPR. И самая важная информация об AMX, безусловно, относится системе команд. Концептуальная блок-схема, относящаяся к реализации AMX, приведена на рисунке 24, и используется здесь для иллюстрации соответствующей части ISA. .

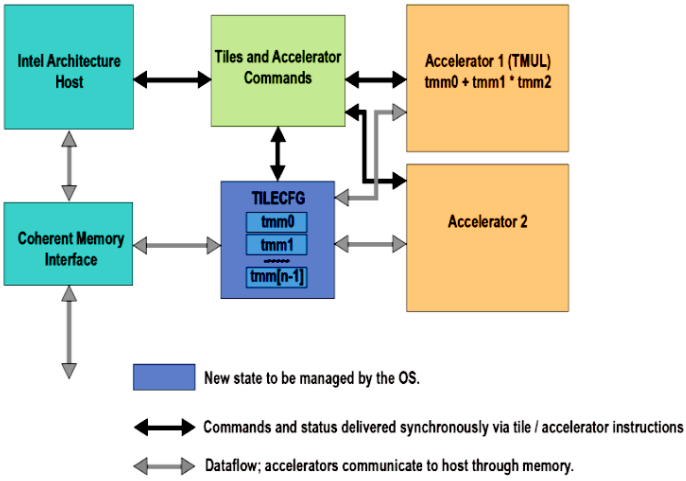


Рисунок 24. Концептуальная схема архитектуры Intel AMX (рисунок из [243])

Подробная информация об AMX, на которой основан дальнейший анализ здесь, имеется в [245]. AMX предназначается для работы с матрицами данных в форматах BF16, FP16 и INT8, актуальных для задач ИИ, и включает две основные компоненты. Это набор двухмерных регистров (плиток), по сути представляющих подмассивы из более крупного двухмерного образа памяти, и акселератор, способный работать с этими плитками. Первая реализация такого акселератора называется TMUL (tile matrix multiply unit) [245]. AMX является расширяемой архитектурой, предполагающей добавление новых акселераторов или форматов данных, или усовершенствование TMUL для увеличения производительности.

AMX перечисляет программисту палитру (pallete) опций, которая указывает, как можно программировать работу с плитками. В настоящее время имеется две палитры, основной для использования AMX является палитра 1. Она включает набор из 8 плиток-регистров с именами TMM0...TMM7. Каждая плитка имеет максимальный размер 16 строк по 64 байта (1 КБ), однако программист может настроить каждую плитку на меньшие размеры, соответствующие используемому им алгоритму [245].

Это естественно соответствует матрице 16×64 с форматом INT8 или 16×32 с форматом BF16. Информацию о максимальном количестве поддерживаемых плиток и максимальном размере плиток для конкретного используемого оборудования можно получить через команду CPUID.

Номер палитры (palette_id) и размеры плиток являются метаданными, которые хранятся внутри еще одного специального регистра управления (TILECFG).Поскольку выполнение управляется метаданными, существующий двоичный файл Intel AMX может использовать преимущества больших размеров хранилища и более производительных блоков TMUL, выбрав самую мощную палитру, указанную CPUID и соответственно корректируя цикл и указатель. TILECFG программируется с помощью команды LDTILECFG. Изначально AMX включала 12 команд, затем их число возросло до 17, в том числе из-за расширения используемых форматов данных. Эти команды осуществляют общую настройку, загрузку/сохранение плиток и собственно матричные операции умножения с плитками. Такие сложные команды, как матричные операции, предполагают много тактов для исполнения в TMUL [**245**]. Данные о количестве тактов и величинах задержки для команд AMX имеются в [**246**].

Для AMX используются и термины, аналогичные применяемым для обычных акселераторов, например, для GPU. Так, часть процессора и используемая им память именуется хостом (см. рисунок 24). Этот термин используется при наличии в серверах обычных акселераторов, в том числе GPU. AMX исходно нацелен на усовершенствования и возможности масштабирования в будущем. На данном рисунке указано два акселератора, хотя в Xeon SPR имеется один TMUL. В будущем возможны улучшения TMUL и/или увеличение числа акселераторов [**245**].

Команды AMX синхронны с потоком обычных команд x86, а команды загрузки/сохранения плиток обеспечивают когерентность относительно памяти хоста. Хост может использовать любые свои ресурсы (например, AVX-512) параллельно с AMX [**245**].

Надо иметь в виду, что возможность работы с AMX обеспечивается только достаточно новыми версиями ядер Linux (от 5.16.20 и выше). Это означает, что поставляемые в часто используемых в настоящее время дистрибутивах (например, RHEL 9 и все совместимые с ним дистрибутивы, SLES 15.4 и соответствующий OpenSUSE, Ubuntu 22.04) версии ядер для AMX не подходят. Но AMX поддерживается ядрами в SLES 15 SP6, Ubuntu 22.10 и выше и Fedora от 36-й версии и выше.

Имеющиеся данные о достигаемой производительности при использовании AMX для задач ИИ на процессорах Xeon SPR рассматриваются далее в разделе 4. Выше уже говорилось о возможных нарушениях безопасности в современных процессорах. В соответствии с [**247**], AMX также может быть использовано для этого.

Расширений в ISA у Xeon SPR много. Часть новых команд сосредоточена на архитектуре интерфейса акселератора (AiA). Они встроены в ISA и представляют собой усовершенствование процессоров x86, предназначенное для оптимизации перемещения данных от акселераторов к сервисам хоста [**243**].

Из других расширений в [**243**] отмечаются новые команды для поддержки виртуализации и обеспечения безопасности. Подробности см. в [**245**, **246**].

3.1.3. *Микроархитектура процессоров Xeon SPR*

Мы начнем анализ микроархитектуры Xeon SPR с общей относящейся к области «вокруг 4-го поколения» процессоров Xeon таблицы 25. Она дополняет данные таблиц 23 и 24 и в целом характеризует изменения количественных показателей данных процессоров, которые во многом и определяют прогресс, достигаемый в соответствующих поколениях. В тексте далее и в таблице 25 больше внимания уделено пропускным способностям из-за их важности для производительности.

Что касается величины адресуемой физической и виртуальной памяти, то для традиционных дающих масштабирование до двух сокетов процессоров Xeon используется 52 и 57 битов соответственно, столько же, сколько и в EPYC 9004. Это применимо и для Xeon SPR в 4- и 8-сокетных конфигурациях (в предыдущем поколении, Cooper Lake-SP, адресуемая память была меньше).

Максимальная теоретическая пропускная способность (B, ГБ/с) памяти и межсоединения между процессорами рассчитывается из данных таблицы 25 по обычной формуле

$$B = N \times S \times W/8,$$

где N – число каналов (памяти или UPI), S – скорость передачи (GT/s для UPI или одна тысячная от MT/s для DDR), W – ширина канала в битах, например, 24 для UPI в Xeon SPR и EMR и 64 для DDR4 или DDR5.

Исходя из данных таблиц 23, 24 и 25, прогресс в Xeon SPR (это соответственно во многом относится и к Xeon Max, см. раздел 3.2) в первую очередь проявился в существенном увеличении числа ядер, для чего применялась и усовершенствованная технология Intel. Но отставание по числу ядер от AMD EPYC, как и относительно используемой для производства EPYC полупроводниковой технологии, сохранилось.

Возросли также емкости кэша L2 и L3. Был реализован переход с DDR4 на DDR5-4800. Учитывая более медленный рост пропускной способности памяти по сравнению с производительностью процессоров, эти усовершенствования в иерархии памяти крайне важны. Был также произведен переход на поддержку PCIe 5.0, и существенно увеличено число линий PCIe.

Таблица 25. Характеристики 3-го, 4-го и 5-го поколений масштабируемых процессоров (МП) Xeon

	МП третьего поколения		МП четвертого поколения Xeon SPR	Xeon Max	МП пятого поколения Xeon EMR
	Xeon ICL	Xeon Cooper Lake-SP			
Число сокетов	1, 2	4, 8	1,2,4,8[1]	1,2	
Технология, нм	10	14	10		
Максимальное число ядер	40	28	60	56	64
I-кэш L1/D-кэш L1	32 КБ/48 КБ	32 КБ/32 КБ	32 КБ/48 КБ		
Кэш L2 (частный для ядра)	1.25 МБ	1 МБ	2 МБ		
Емкость кэша L3 (максимум)	60 МБ	38.5 МБ	112.5 МБ		320 МБ
TDP (макс.)	270 Вт	250 Вт	350 Вт		
Бит физического/ виртуального адреса	52/57	46/48	52/57		
Число контроллеров памяти/кластеров Sub-Numa	4/2	2/2	4/4		4/2
Тип памяти	DDR4		DDR5		
Скорость передачи,MT/s	3200 (2DPC)	3200 (1DPC) 2933 (2DPC)	4800 (1DPC) 4400 (2DPC)		5600 (1DPC) 4800 (2DPC)
Максимальное число каналов памяти	8	6	8		
Размер памяти HBM2E	нет			64 ГБ	нет
Максимальное число каналов UPI	3×20	6×20	4×24		
Максимальная скорость UPI	11.2 GT/s	10.4 GT/s	16 GT/s		20 GT/s
Версия PCIe	PCIe v4		PCIe v5		
Максимальное число линий PCI	64	48	80		

[1] Более 8 реализуется через поддержку xNC (см. в разделе 3.1.4). Данные из [49, 50, 238, 243, 248].

Общая блок-схема, иллюстрирующая микроархитектуру Xeon SPR, представлена на рисунке 25. Все блоки на этом рисунке понятны по их названию, а Ch0 и Ch1 на краях рисунка означают каналы памяти.

В Xeon SPR впервые стало использоваться межсоединение Intel EMIB. Оно преодолевает ограничение одной сетки и обеспечивает более эффективное масштабирование по сравнению с монолитными конструкциями. EMIB применяется Intel и для других аппаратных средств, например, для работы с FPGA. Вообще EMIB давно и хорошо известно по целому ряду публикаций [249–252], а в Xeon SPR было проведено соответствующее усовершенствование [238].

Рисунок 25 демонстрирует наличие до четырех чиплетов в разных моделях Xeon SPR. Их применение, как видно из этого рисунка, означает наличие NUMA уже в рамках одного процессора, поскольку у каждого чиплета имеются своя близкая память и соответствующие ее контроллеры Ch0/Ch1. На самом деле такая возможная локализация охватывает не только память, но и кэш LLC.

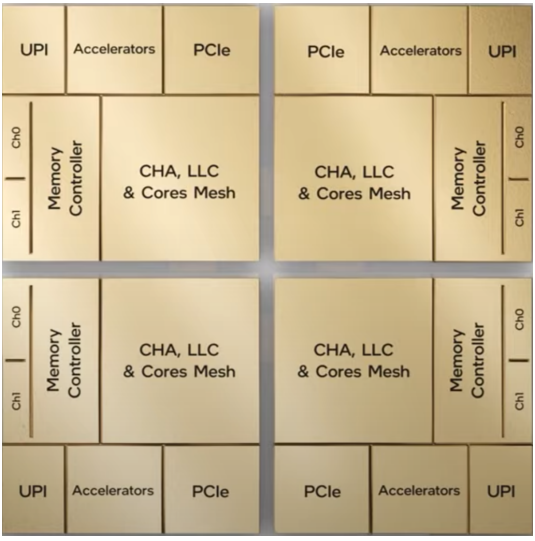


Рисунок 25. Основные блоки микроархитектуры Xeon SPR (рисунок из [253], соответствует представленному Intel в [239])

Intel использует для памяти процессоров Xeon SPR термин «домен унифицированного доступа к памяти» UMA (Uniform Memory Access) [243]. Он обеспечивает единое адресное пространство, которое чередуется между всеми контроллерами памяти. Но из блоков общей для всего процессора памяти и близких к ним чиплетов можно образовать NUMA-кластеры

SNC, о которых уже говорилось выше в начале раздела 3. Учет NUMA благодаря применению доменов SNC позволяет эффективно использовать более низкие величины задержек обращения к локально расположенной части LLC и близкой памяти. В соответствии с рисунком 25 можно логически разбить процессор на два (SNC2) или на четыре кластера (SNC4).

SNC дает уникальное местоположение для каждого адреса в LLC, и в доменах LLC оно не дублируется. Локализация адресов в LLC для каждого домена SNC применяется только к адресам, сопоставленным контроллерам памяти в том же сокете. Все адреса, сопоставленные с памятью удаленных сокетов, равномерно распределяются по всем банкам LLC независимо от режима SNC. При использовании режима SNC каждое ядро, как обычно, имеет доступ ко всему LLC процессора.

При разбиении на четыре SNC получается «наиболее тонкий» учет NUMA-особенностей; этот вариант изображен ниже на рисунке 26.

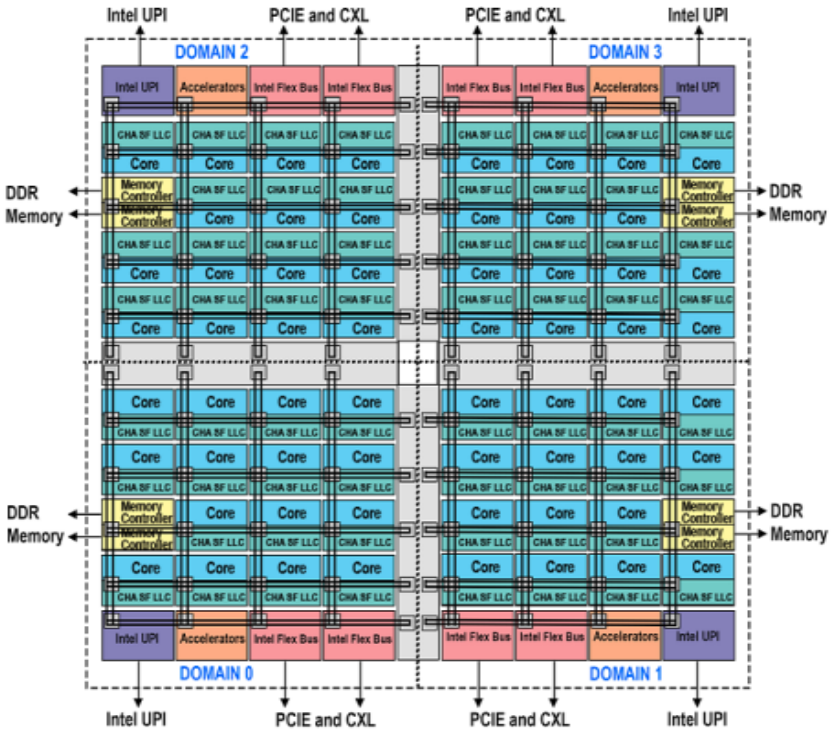


Рисунок 26. Блок-схема Xeon SPR с четырьмя SNC-доменами (рисунок из [243])

Рисунок 26 является более детализированным, чем рисунок 25. В нем отображена также сетка, используемая как межсоединение для ядер процессора [243].

Естественно, практическое применение SNC может потребовать модернизацию текста использующего SNC программного обеспечения, обычно с применением OpenMP. Но Intel предложила еще два интересных варианта возможного учета NUMA-особенностей без программных доработок. Такие варианты с разбиением процессора на 2 и 4 части называются соответственно полушарие и квадрант.

Процессор имеет доступ ко всем агентам кэширования и контроллерам памяти. Однако на основе хэш-функции каждый домашний агент автоматически обеспечивает маршрутизацию агента кэширования на ближайший контроллер памяти в том же домене. Как только процессор обращается к агенту кэширования, расстояние сводится к минимуму, поскольку он всегда будет взаимодействовать с ближайшим контроллером памяти [243]. Как уже было отмечено выше в начале раздела 3, агенты кэширования и домашние агенты объединены в блоках CHA. Эти два режима работы с хэш-функциями могут давать более высокую производительность относительно UMA, но, естественно, не так хорошо, как настоящая кластеризация SNC. По умолчанию в BIOS установлен как раз режим работы с квадрантом.

Эти режимы, как и настоящая кластеризация SNC, требуют симметричного расположения памяти. Если симметрия имеющейся памяти недостаточна для режима квадранта, будет применяться режим полусферы. Если же и для него симметрия памяти не подходит, то будет работать UMA [243]. Безотносительно к вариантам формирования SNC и память, и LLC остаются при этом общими для всего процессора, просто появляется возможность «приоритетной» работы с близкими компонентами LLC и памяти.

Акселераторы. В разделе 3.1.2 уже говорилось про архитектуру интерфейса акселератора AiA и про самый актуальный для задач ИИ акселератор AMX. Кроме него, Xeon SPR могут иметь еще 4 других типа акселераторов Intel: IAA (In-memory Analytics Accelerator), DSA (Data Streaming Accelerator), DLB (Dynamic Load Balancer) и QAT (QuickAssist Technology). Нижеследующая краткая информация базируется на [243].

IAA может ускорить работу с помощью примитивов аналитики (сканирование, фильтрация и т. д.), сжатия разреженных данных и ранжирования памяти (memory tiering). Это важно для работы с большими данными и с БД в памяти.

Ранжирование памяти предполагает разделение памяти на разные области для облегчения управления ресурсами. Часто используемые (горячие) данные могут находиться в области, где пропускная способность

для процессора более высока, а менее интенсивно используемые (холодные) данные могут находиться в области с более медленной пропускной способностью. Горячие данные могут храниться в памяти, а холодные данные – на жестком диске.

IAA может иметь до четырех экземпляров на сокет, которые доступны для выгрузки данных из ядра процессора. С точки зрения программного обеспечения каждый экземпляр представляет собой сложную интегрированную конечную точку PCIe. IAA также поддерживает общую виртуальную память, позволяя устройству работать непосредственно в виртуальном адресном пространстве приложений.

DSA заменяет более старую технологию Intel Quick Data. Он помогает разгрузить общие операции перемещения данных из ядер процессора, которые могут быть обнаружены в приложениях хранения, гипервизорах, операциях с постоянной памятью (с Intel Optane), сетевых приложениях, очистке кэша операционной системы, обнулении страниц и перемещении памяти.

DSA имеет до четырех экземпляров на сокет в зависимости от модели. Он поддерживает перемещение данных между любыми компонентами платформы, включая всю совместимую подключенную память и устройства ввода-вывода. Например, поддерживается перемещение данных между кэшем процессора в основную память, между кэшем процессора и дополнительной картой и между двумя дополнительными картами.

Поскольку в современных вычислительных системах часто происходит отказ от применения HDD с переходом на работу с оперативной памятью (например, на системы баз данных в памяти), и актуальным может стать применение CXL-памяти, важность DSA потенциально может увеличиться. Работа DSA с CXL рассмотрена, например, в [**254**].

Другие возможности DSA возникли из-за применения двух уровней памяти (HBM и DDR) в Xeon Max, использующих микроархитектуру Xeon SPR. Данные о большом увеличении пропускной способности за счет применения DSA в 2P-сервере с 48-ядерными Xeon Max 9468 при работе с одной или несколькими нитями, использовании одного или нескольких DSA, с применением одного или двух сокетов получены в [**255**].

В соответствии с [**256**], на что в качестве примера возможного эффективного применения DLB указано в документации Intel по Xeon SPR [**243**], DLB – это устройство PCIe, которое обеспечивает сбалансированное по нагрузке и управлению приоритетами планирование событий (пакетов) по ядрам/потокам процессора. Этот ускоритель в Xeon SPR расположен внутри процессора.

В [**256**] указывается на применение DLB с помощью набора библиотек с открытым исходным кодом DPDK (Data Plane Development Kit), хотя его использование не является необходимым. Оптимизации работы с системой

очередей, для чего применим DLB, с точки зрения [256] актуальна из-за роста активности применения акселераторов. Другой возможной областью использования DLB там указываются задачи оптимизации трафика в телекоммуникационных задачах.

QAT предназначен для ускорения решения криптографических задач. Но если предыдущие поколения QAT располагались в чипсете, то затем QAT перенесен в корпус процессора. В QAT в Xeon SPR были увеличены скорости шифрования, расшифровки RSA и сжатия данных.

С другой стороны понятно, что использование акселераторов требует не только применения драйверов Intel, но и специального программного обеспечения. Ориентация на работу с ними вызывает непереносимость соответствующего кода на другие аппаратные платформы.

С нашей точки зрения потенциально более важным, чем применение в Xeon SPR акселераторов (кроме AMX), может оказаться наличие в этих процессорах поддержки CXL 1.1 [243], которая дает возможность подсоединения CXL-памяти как возможного экономически эффективного пути сглаживания разрыва между основной памятью и внешней. Постоянная память (в первую очередь Intel Optane), учитывая и отказ Intel от ее дальнейшего производства, этой проблемы не решила.

TDP и изменение частоты. Максимальные значения TDP у моделей Xeon SPR в сравнении с другими поколениями масштабируемых процессоров Xeon приведены в таблице 25, а в таблицах 23 и 24 TDP для старшей модели Xeon 8490H сопоставлены с данными для старших моделей EPYC 9004. Хотя TDP для EPYC немного выше, число ядер в старших моделях EPYC гораздо выше. Определенное отставание в полупроводниковой технологии Intel 7 от используемой для производства EPYC технологии от TSMC, безусловно, влияет и на TDP.

Представленные в таблицах 23 и 24 старшие модели EPYC Milan и EPYC Genoa имеют более высокие базовые и турбо-частоты, чем Xeon; 96-ядерный EPYC 9654 также имеет более высокую турбо-частоту всех ядер, чем Xeon 8490H.

Понятно, что Intel прилагает большие усилия по оптимизации энергопотребления и управлении применяемой тактовой частотой. В процессорах Intel Xeon SPR введены новые оптимизированные по энергопотреблению C-состояния C0.1 и C0.2 (они являются подсостояниями C0). Как и более глубокие C-состояния (C1, C1E, C6), выполнение инструкций останавливается, как только ядро переходит в одно из этих новых C-состояний. Новые C-состояния имеют гораздо меньшие задержки выхода, что важно для ускорения работы в обычных сетях, например, с использованием библиотеки DPDK [257].

Что касается собственно регулирования частот, это имеет отношение к P-состояниям. Эти состояния в Xeon SPR могут давать снижение рабочего напряжения и частоты ядер и соответственно уменьшение потребляемой энергии. Включение или отключение технологии Intel Turbo Boost влияет, естественно, на P- и C-состояния (подробнее см., например, [**258**]). Для достижения максимальной производительности Turbo Boost практически всегда включается, но его можно отключить для достижения более низкой, но предсказуемой производительности.

Ядра Linux для Xeon SPR имеют два разных драйвера. Во-первых, это традиционный драйвер acpi-cpufreq, используемый подсистемой CPUfreq. Имеющиеся в ней разные регуляторы описаны выше в разделе 1.2, а их применение в EPYC Zen 4 – в разделе 2.2.

Другой драйвер, intel_pstate, реализует два своих собственных алгоритма. Алгоритм «performance», реализованный intel_pstate, похож на одноименный алгоритм в CPUfreq. Алгоритм алгоритм «powersave» в intel_pstate больше похож на «schedutil» в CPUfreq. По умолчанию загружается драйвер intel_pstate. Для vCMTS (virtualized cable modem termination system) в рекомендуется работать с регулятором «userspace» в традиционным драйвером, поскольку это позволяет иметь полный контроль над настройкой частоты [**257**].

В командной строке Linux с помощью инструмента python power.py[33], доступного в репозитории Intel CommsPowerManagement GitHub, можно отслеживать и изменять параметры P-состояний.

В практическом плане актуальным усовершенствованием в Xeon SPR надо отметить появление в BIOS специального режима «Optimized Power Mode» (OPM), который позволяет иметь высокую производительность при заметном понижении энергопотребления (см., например, [**259**]). Иллюстрация эффективности его применения приведена далее в разделе 4.1.

Виртуализация. Рассмотренные выше особенности применения NUMA в Xeon SPR, безусловно, отражаются в реальном использовании средств виртуализации, поскольку это способствует естественному образованию виртуальной машины на каждом NUMA-узле.

Так, например, знаменитый в бизнес-сфере программный продукт SAP HANA, который раньше обеспечивал работу на 2P-сервере двух виртуальных машин, теперь поддерживает работу с виртуальными машинами в режиме SNC2, по две виртуальные машины на процессор и соответственно четыре – на сервер [**260**].

Intel вообще отличается огромным набором областей, в которых фирма достигает успехов. В области виртуализации вообще можно упомянуть vRAN (развертывание сетей виртуального радиодоступа), для чего Intel

[33] https://github.com/intel/CommsPowerManagement, accessed 18.11.2025.

создала специальные средства vRAN boost, и специализированные Xeon Sapphire Rapids EE. Однако все это совсем не подпадает в тематику данного обзора.

Технология виртуализации Intel VT-X, включающая и соответствующие команды в ISA, VMX (virtual-machine extensions), существует очень давно и сильно развились за многие годы. В VT-X также очень давно используется технология виртуализации таблиц страниц EPT (Extended Page Tables). К настоящему времени VT-X имеет очень много возможностей. Многие из них имеют аналоги в процессорах Zen-4, и были кратко рассмотрены выше. Здесь мы остановимся только на последних усовершенствованиях в этой области, имеющихся в Xeon SPR, которые относятся к VT-d (Virtualization Technology for Directed I/O), добавляющей новый уровень аппаратной поддержки виртуализации устройств ввода/вывода [**261**].

VT-d использует, в частности, виртуализацию межпроцессорных прерываний IPI (Inter-Processor Interrupts). IPI – это часть технологии виртуализации прерываний Intel APICv, аналогом которого является AMD AVIC. IPI стабильно поддерживается только новыми версиями ядер Linux (от 5.19 и выше).

Intel в своем основном техническом документе о Xeon SPR [**243**] указывает на появившуюся в этих процессорах свою новую технологию Scalable IOV, заменившую ранее существовавшую технологию виртуализации ввода-вывода SR-IOV, предоставив гибкую композицию виртуальных функций с использованием программного обеспечения собственных аппаратных интерфейсов.

Если SR-IOV реализует полный интерфейс виртуальных функций, Scalable IOV вместо этого использует облегченный интерфейс, оптимизированный для быстрых операций с путями передачи данных для прямого доступа со стороны гостя. Intel Scalable IOV легко взаимодействует с PCIe или CXL через идентификаторы адресного пространства процесса, позволяя нескольким гостевым операционным системам одновременно использовать устройства PCIe или CXL. Intel Scalable IOV поддерживает также все акселераторы, присутствующие в Xeon SPR, и любые дискретные акселераторы. Но Scalable IOV требует поддержки со стороны операционных систем и диспетчеров виртуальных машин [**243**].

Безопасность. В Xeon SPR для аппаратной поддержки безопасности применяется новая технология Intel, TDX (Trust Domain Extensions). Средства обеспечения безопасности в EPYC Zen 4 вкратце рассматривались выше в разделе 2.2.

Исследования средств безопасности всех современных процессоров являются активно развиваемой научной областью. Эти средства ограниченным образом относятся к производительности из-за их редкого применения, например для HPC. Они соответственно не рассматриваются

в разделах обзора, где сопоставляется производительность анализируемых в нем процессоров. Поэтому небольшое сравнение этих аппаратных средств рассматриваемых процессоров целесообразно провести здесь.

Во введении уже отмечалось, что атаки, приводящие к нарушениям безопасности, возможны для почти всех современных суперскалярных процессоров. Укажем здесь на еще один вариант атаки для таких процессоров, Downfall, и соответствующую публикацию [**262**]. Как отмечено в [**262**], Intel утверждает, что Xeon SPR этим не затронуты. В AMD Zen 2 предварительные тесты не выявили утечки данных, но авторы [**262**] планируют продолжить исследования процессоров Intel и других производителей.

Рассматриваемые здесь аппаратные средства обеспечения безопасности ориентированы в первую очередь на виртуализацию и облачные технологии. Intel, предложив для своих процессоров средства безопасности SGX (Software Guard Extensions) раньше других производителей, в 2015 году, сформулировала в качестве основы своей парадигмы конфиденциальных вычислений четкую ориентацию: те, кто контролирует платформу, и те, у кого есть обрабатываемые данные на платформе, являются двумя отдельными сущностями. В то время как обычно владелец платформы имеет полный доступ к тому, что обрабатывается на платформе [**263**].

Виртуализация и применение ее для облачных технологий, естественно, поднимают вопросы обеспечения безопасности на гораздо более более важный уровень. В качестве иллюстрации укажем, что, по данным IBM, 82% утечек данных относились к хранящимся в облаке данным [**264**].

Для самых разных технологий обеспечения безопасности общим является термин TEE (Trusted Execution Environment). Средства для TEE должны гарантировать конфиденциальность и защищенность от несанкционированного доступа, а также быть невосприимчивыми к различным типам атак. Выше уже было отмечено, что гарантии этого пока достаточно сомнительны, поскольку появляются данные о все новых нарушающих безопасность вариантах атак. Поэтому средства для TEE обеспечивают скорее только определенные типы и уровни безопасности. Средства безопасности в процессорах ARM, и SEV в AMD EPYC, и Intel SGX относятся к TEE.

К важнейшим показателям аппаратных средств для TEE относится не только достигаемый уровень обеспечения безопасности, но и то, насколько эти средства могут вызывать понижение достигаемой производительности. В качестве примера работы, посвященной влиянию их на производительность для HPC, укажем [**265**]. И необходимо сразу отметить, что применение и Intel SGX, и AMD SEV продемонстрировало в [**265**] возможное очень сильное понижение производительности.

Intel SGX базируется на изоляции и шифровании частей кода, которые обрабатывают конфиденциальные данные в приложении. Данные шифруются и дешифруются с помощью хорошо известных криптографических алгоритмов налету (перед записью в память системы или при чтении из нее). Немного иная стратегия – это изоляция и шифрование памяти всей виртуальной машины, что используется в AMD SEV [**266**].

Средства SGX с точки зрения достигаемого уровня безопасности нередко считались лучшими – см., например, [**264**, **267**]. Однако известна, например, атака CipherLeaks, в которой нарушается защита и SGX, и SEV (см., например, [**268**]).

В качестве недостатка применения SGX часто указывают, что это требует модификации кода приложений и возлагает на их разработчиков ответственность за написание устойчивого к атакам кода [**266**], и поэтому его считают сложным для использования [**264**].

Но причиной отказа Intel от SGX и перехода в Xeon SPR на применение технологии TDX (Trust Domain Extensions) было, скорее всего, ограничение размера безопасной памяти [**264**]. Это делало SGX неприемлемым для задач HPC [**265**].

AMD, использовавшая в SEV (с 2016 года) стратегию на основе изоляции памяти всей виртуальной машины, улучшала SEV уже дважды: в 2017 году до SEV-ES (Encrypted State), а в 2020 году – до SEV-SNP (Secure Nested Paging) [**266**]. Эти технологии кратко рассмотрены ранее в разделе 2.2.

Появившаяся в Xeon SPR новая технология Intel TDX, как и AMD SEV, позволяет развертывать аппаратно-изолированные виртуальные машины, которые в TDX называются доменами доверия (Trust Domain, TD). Дальнейшее рассмотрение TDX здесь основано на [**263**] и последнем на момент подготовки обзора описании TDX [**269**].

TD, новый тип гостевой виртуальной машины, расширяет имевшиеся ранее возможности VMX (упоминавшегося выше расширения ISA для VT-X) и многоключевого полного шифрования памяти MKTME(Multi-Key Total Memory Encryption), обеспечивая аппаратную изоляцию виртуальных машин также от среды хостинга. Даже при полном контроле над механизмами управления хостом со стороны монитора виртуальных машин VMM /гипервизора нельзя получить доступ к личной информации виртуальных машин, как это было раньше при виртуализации в облачном хостинге. Отметим, что MKTME встроен в контроллер памяти [**38**].

Программный компонент, который управляет TD, называется модулем TDX. В Xeon SPR в TDX появился новый для x86 режим безопасного арбитража SEAM (SEcure Arbitration Mode), который нужен для изоляции модуля TDX от всех программно-аппаратных компонент, не включенных в базу доверенных вычислений TD, TCB (Trusted Computing Base). Xeon SPR с TDX разрешает доступ к диапазону памяти SEAM только

программному обеспечению, выполняющемуся внутри диапазона памяти SEAM, а все другие программные доступы и прямой доступ к памяти (DMA) с устройств к этому диапазону памяти прерываются.

В диапазоне памяти SEAM обеспечивается криптографическая защита конфиденциальности с использованием AES в режиме XTS со 128-битным ключом шифрования памяти. Может применяться один из двух доступных режимов защиты целостности памяти. Она может быть обеспечена либо (по умолчанию) криптографической схемой защиты, либо использовать схему защиты логической целостности.

Схема криптографической целостности использует 28-битный код аутентификации сообщений (MAC) на основе криптографической хеш-функции SHA-3, а схема защиты логической целостности предназначена только для предотвращения доступа к программному обеспечению хоста/системы (там используется один бит владения TD).

В результате рабочая нагрузка получает возможность применять SEAM (независимо от используемой ею облачной инфраструктуры) для обеспечения более безопасного доступа к разным программно-аппаратным компонентам [**38**, **263**].

Кроме того, TDX включает средства удаленной аттестации, позволяющей проверяющей стороне (например, владельцу рабочей нагрузки) установить, что рабочая нагрузка выполняется на платформе с поддержкой TDX, расположенной в TD, еще до предоставления данных этой рабочей нагрузки.

Данное изложение здесь, как и [**38**, **263**], основаны на TDX 1.0. Intel представила соответствующие ей подробные данные для проведения исследовательских испытаний Google, Microsoft и NCC Group (знаменитой занимающейся кибербезопасностью компании) еще до начала поставок Xeon SPR. Они изучали возможности TDX на предмет соответствия требованиям безопасностью, и предоставили Intel полученные данные, на основе которых фирма провела и необходимые усовершенствования.

Наиболее известным из этих испытаний стало, вероятно, исследование в Google, проводившееся в течение 9 месяцев, подробные результаты которого были опубликованы в [**38**]. Там был проверен 81 вектор атак, что дало, в частности, около 10 уязвимостей безопасности. Согласно [**269**], они все были устранены в выпускаемых Intel Xeon SPR, где применяется уже более новая версия TDX.

Эти данные, с одной стороны, говорят вообще об очень высокой сложности такой реализации средств безопасности. Но Intel получила в результате таких исследований большое преимущество. В [**38**] отмечается также сложность самой системы TDX.

Важно также, что TDX быстро стала доступна в известных дистрибутивах Linux, например, в RHEL 8.10, SLES 15 SP5 или Ubuntu 24.04 LTS.

Как и другие обсуждавшиеся средства обеспечения безопасности, TDX ориентирован на задачи виртуализации и облачных технологий, и не может решить вообще все потенциально возможные проблемы безопасности процессоров. На рисунке 27 дается иллюстрация того, какие области безопасности TDX охватывает, а какие – нет.

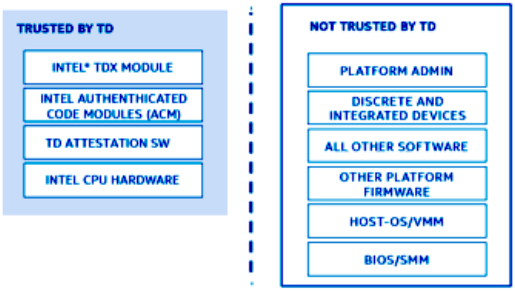


Рисунок 27. Границы доверия TDX (рисунок из [**263**])

В сравнительных оценках разных технологий для TEE надо учитывать и такие «побочные» эффекты: уменьшение производительности, необходимость поддержки в операционной системе и модернизации кодов использующих их приложений. В современных работах предлагаются и вообще новые варианты обеспечения безопасности (см, например, [**264**, **270**]). Все это требует дальнейших исследований, и разумным остается и просто тщательное сопоставление имеющихся возможностей технологий для TEE, как это делалось ранее в [**267**].

3.1.4. *Вычислительные системы на базе Xeon SPR*

Прежде чем перейти от рассмотрения процессоров к вычислительным системам, нужно сказать про чипсет. Хотя Intel относит Xeon SPR к SoC, чипсет (C741) для этих процессоров, в отличие от EPYC 9004, используется. Про C741 уже говорилось выше в разделе 1.1. Наличие этой микросхемы чуть добавляет стоимости ($70) и величины TDP (11 Вт) [**45**].

Традиционные содержащие до двух сокетов системы на базе Xeon SPR предлагаются всеми основными производителями серверов, и в архитектурном плане здесь не возникает ничего особо нового. Отметим только, что использование NUMA предполагает аккуратное конфигурирование по отношению к расположению модулей памяти. Полезным практическим руководством для двухсокетных серверов могут быть рекомендации Lenovo для серверов с процессорами Xeon SPR и Xeon EMR [**271**].

Серверы с 4 или 8 сокетами выпускают Supermicro и Lenovo. Аналогов таких серверов с AMD EPYC нет. Принципиально важным является также то, что на базе процессоров Xeon SPR можно строить системы с общем полем памяти, содержащие больше 8 сокетов – это возможно с применением технологии xNC (eXternal Node Controller), разработанной Intel совместно с Numascale [**272**].

Знаменитыми вычислительными системами, использующими возможности xNC, раньше были HPE Superdome Flex. Аналогичные системы на базе Xeon SPR, HPE Compute Scale-up 3200 Server, имеют модульную архитектуру, применяющую 4-сокетные строительные блоки 5U. Архитектура этих начавшихся поставляться в 2023 году систем представляет собой комбинацию сервера Superdome Flex, который использовал xNC, и сервера HPE SuperdomeFlex 280, в котором применяется бессвязная инфраструктура с расширенным подключением UPI [**273**].

Система Compute Scale-up 3200 обеспечивает масштабируемость от 4 до 16 сокетов с шагом в 4 сокета и масштабирование DDR5 до емкости 32 ТБ. Подробнее архитектура этих серверов описана в [**274**]. Они предназначены в первую очередь для задач SAP HANA и работы с базами данных в памяти.

Суперкомпьютеры в июньском списке TOP500 2024 года содержат Xeon SPR как в гетерогенных узлах с GPU, так и в гомогенных узлах только с процессорами.

В качестве наиболее известного американского суперкомпьютера с Xeon SPR, содержащего в узлах еще Nvidia H100, следует указать Microsoft Eagle ND v5 [**275**], занимающую третье место в списке виртуальную машину Microsoft ND v5 из семейства Azure, работающую с Ubuntu 22.04. Там в узлах используются 56-ядерные Xeon 8480C. Суффикс C в названии SKU в общем списке суффиксов на рисунке 20 отсутствует. Возможно, это предварительный вариант Xeon 8480+. Там применяются такие же ядра с базовой тактовой частотой два ГГц, но максимально допустимая температура составляет 73 градуса Цельсия, а в Xeon 8480+ она повышена до 79 градусов.

В занимающем седьмое место в этом списке европейском суперкомпьютере Leonardo есть называемая «data-centric» часть (далее мы используем для таких частей-кластеров термин раздел) с гомогенными узлами на базе Xeon 8480+, но он дополняется разделом «booster», где в узлах применяются GPU [**276**].

Занимающая в списке 22-е место вычислительная система Mare Nostrum 5 GPP в известном барселонском суперкомпьютерном центре [**277**] использует гомогенные узлы GPP. Их имеется 3 типа: обычные, с расширенной памятью и с памятью HBM. В обычных узлах и в узлах с расширенной памятью применяется по два 56-ядерных Xeon 8480+ с памятью DDR5 по 256 ГБ и 1 ТБ соответственно. В узлах с HBM используются также 56-ядерные Xeon CPU Max (см. в разделе 3.2 далее). В качестве серверов MareNostrum 5 GPP использует Lenovo ThinkSystem SD650 V3.

Однако кроме гомогенных узлов, в MareNostrum 5 имеются еще гетерогенные узлы, содержащие по два 40-ядерных Xeon 8460Y+/2.3 ГГц

и по 4 GPU H100. В целом MareNostrum 5, использующий межсоединение InfiniBand NDR200, является пре-экзафлопсным суперкомпьютером совместного предприятия EuroHPC [277].

Данные о производительности вычислительных систем с процессорами Xeon SPR рассмотрены далее в разделе 4, а в следующем разделе анализируется модифицированное семейство процессоров Xeon SPR, в котором добавлена поддержка работы с HBM-памятью, в том числе совместно с DDR-памятью. Соответственно содержащие в узлах Xeon SPR с HBM суперкомпьютеры, такие как Crossroads [278], также рассматриваются далее.

3.2. Процессоры Xeon Max (Xeon SPR с HBM)

Сначала в публикациях процессоры Xeon Max вообще называли Xeon Sapphire Rapids с памятью HBM. Главное, чем Xeon Max отличаются от Xeon SPR, это наличие в каждом из четырех чиплетов по одному контроллеру высокоскоростной памяти HBM2e в дополнение к обычным контроллерам DDR5. Максимально возможное число ядер в Xeon Max при этом уменьшилось на четыре, до 56.

Память HBM, как известно, состоит из нескольких стеков DRAM-памяти и использует большую ширину шины, обеспечивающую высокую пропускную способность. Хотя HBM-память обычно используется в современных GPU [2], она применяется также и в знаменитых ARM-процессорах Fujitsu A64FX [7]. HBM-память дорогая, и ее емкость в GPU или в A64FX фиксирована, как и в Xeon Max.

В Xeon Max имеется четыре кристалла HBM – стеки, содержащие по 8 модулей DRAM. К каждому из четырех контроллеров памяти HBM подсоединяется один кристалл HBM емкостью 16 ГБ [280] (см. рисунок 28). Соответственно общая емкость HBM в каждой модели Xeon

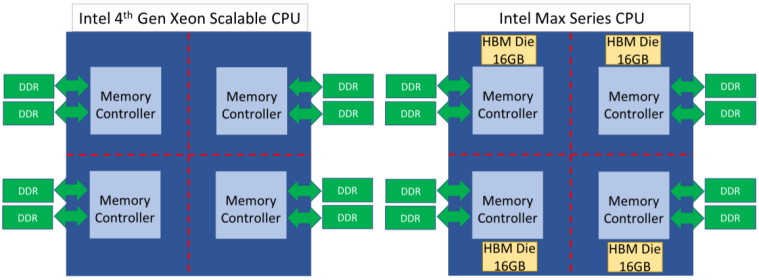


Рисунок 28. Сопоставление архитектур Xeon SPR и Xeon Max (рисунок из [280])

Max составляет 64 ГБ. При ширине шины 1024 бита и скорости 3.2 GT/s четыре стека могут обеспечить пиковую пропускную способность 1638.4

ГБ/с. Но вследствие особенностей расположения отдельных процессорных ядер в решетке Xeon Max (см. рисунок 26) при работе с HBM их пиковая пропускная способность может отличаться [**281**]. В [**279**] указано, что максимальная пропускная способность этой памяти в Xeon Max составляет 1 ТБ/с (реально достигаемая пропускная способность рассматривается ниже в разделе 4.4). Память DDR5 у моделей Xeon Max может применяться дополнительно к HBM или отсутствовать.

Все модели Xeon Max (число разных SKU гораздо меньше, чем у Xeon SPR) применяют номера SKU вида 94xx без наличия суффиксов. Хотя число ядер в разных SKU Xeon Max бывает от 32 до 56, все они обладают HBM-памятью одинаковой емкости 64 ГБ и имеют большое число других совпадающих характеристик, в том числе одинаковыми величинами турбо-частот (3.5 ГГц) и TDP, равной 350 Вт. Все SKU Xeon Max могут работать на серверах, содержащих до двух процессоров. Поддержка работы с Intel Optane в этих процессорах отсутствует [**279**]. Характеристики этих SKU приведены в таблице 26.

Таблица 26. Характеристики моделей Xeon Max

Номер модели	Число ядер	Базовая частота	Размер кэша L3	Число каналов UPI (максимальное)	Цена
9462	32	2.7 ГГц	75 МБ	3	$7995
9460	40	2.2 ГГц	97.5 МБ	3	$8750
9468	48	2.1 ГГц	105 МБ	4	$9900
9470	52	2 ГГц	105 МБ	4	$11590
9480	56	1.9 ГГц	112.5 МБ	4	$12980

Данные из [https://ark.intel.com/content/www/us/en/ark/products/series/232643/intel-xeon-cpu-max-series.html 11.08.2024].

Процессоры Xeon Max поддерживают только конфигурации 1P или 2P. Другие данные об этих процессорах приведены выше в таблицах 23, 24 и 25. Учитывая описанные выше изменения в Xeon Max по сравнению с Xeon SPR, далее рассматриваются только особенности режимов работы с памятью.

Дальнейшее рассмотрение в данном разделе основано на информации [**279**]. Всего в Xeon Max имеется 3 базовых режима работы с HBM.

HBM-only. В качестве базового можно рассматривать режим HBM-only, в котором DDR5 вообще не используется. Это, естественно, предполагает работу с приложениями, для которых емкость 64 ГБ является достаточной. Такие приложения могут получить повышенную производительность благодаря более высокой пропускной способности HBM.

Поскольку у каждого чиплета есть свой контроллер HBM, в этом режиме также может быть достигнуто повышение производительности за счет использования средств NUMA.

Выше в разделе 1.2 обсуждалось применение стандарта ACPI по отношению к частотам процессоров. Также относящаяся к ACPI статическая таблица атрибутов гетерогенной памяти (содержашая, в частности, задержку и пропускную способность, для Xeon Max – для HBM и DDR), как указано в [**279**], используется ядром Linux для оптимизации работы с NUMA. Это позволяет Linux закрепить приложение на определенном узле NUMA, связанном с HBM, и назначить ему приоритет. Поддержка такого NUMA-режима обсуждается ниже в обсуждении режимов кластеризации.

Плоский режим Flat Mode (другое название – одноуровневая память, 1LM). Этот режим можно использовать, если емкости HBM-памяти не хватает, и в дополнение к ней применяется DDR5. HBM и DDR настраиваются как основная память в разных адресных пространствах NUMA. В этом случае требуется не только настройка BIOS (UEFI), но и привязка приложения к соответствующему домену NUMA с помощью специальных программных инструментов NUMA.

Режим кэширования Cache Mode (другое название – двухуровневая память, 2LM). В этом режиме, в отличие от плоского режима, HBM невидим для операционной системы и приложений, поскольку HBM действует здесь как кэш уровня четыре для оперативной памяти DDR, а не как основная память. В этом режиме не возникает необходимости настраивать под него приложения, но по сравнению с плоским режимом достигаемая производительность может быть меньше.

Режимы кластеризации. Поскольку и память HBM, и DDR5 имеют собственные контроллеры в каждом из четырех чиплетах процессора, приведенные выше 3 базовых режима могут использовать более тонкую настройку NUMA с использованием описанных ранее возможностей SNC. Такие режимы кластеризации описаны в руководстве Intel по конфигурированию и настройке Xeon Max [**282**]. Детальное рассмотрение этих режимов, включая графические иллюстрации и рассмотрение также 2P-конфигураций имеется также в руководстве Lenovo к своим серверам [**280**]. В [**280**, **282**] приводится также информация о работе с этими режимами в Linux.

Для каждого из трех базовых режимов работы HBM в Xeon Max поддерживается еще по два режима кластеризации – с применением SNC4 или квадранта, что дает в результате 6 разных режимов.

При использовании кластеризации с квадрантами в режиме HBM-only два разных узла NUMA образуются только в 2P-сервере, а в сочетании с плоским режимом в таком сервере будет уже четыре узла NUMA, поскольку в каждом сокете один узел NUMA есть для DDR5, а один – для HBM. При работе с квадрантами в режиме кэширования память HBM невидима, и на два сокета будет также два узла NUMA.

Использование не квадрантов, а SNC4 дает возможности достижения более высокой производительности (с точки зрения пропускной способности и задержки) при работе с памятью, поскольку в этом случае учитывается наличие четырех разных контроллеров и соответствующих кристаллов HBM по 16 ГБ для каждого процессора. Наиболее высокую производительность можно получить в сочетании SNC4 с режимом HBM-only (см. рисунок 29), что для двух сокетов дает 8 узлов NUMA. Это, возможно,

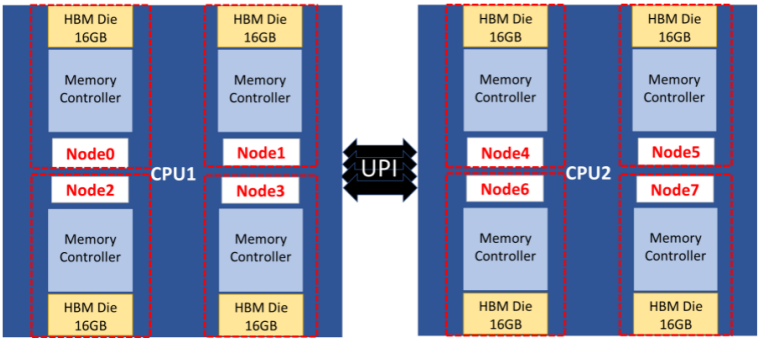


Рисунок 29. Режим HBM-only и SNC4 (рисунок из [280])

требует соответствующих модернизаций в приложении и предполагает, что в 2P-сервере 128 ГБ HBM для приложения достаточно.

Если этой емкости недостаточно, и используется еще DDR5, то максимизацию производительности возможно получить в сочетании SNC4 с плоским режимом. Тогда и HBM, и DDR будут разделены на четыре независимых узла NUMA в каждом Xeon, а в двухсокетном сервере будет шестнадцать узлов NUMA.

Наконец, возможно сочетание SNC4 с режимом кэширования. Поскольку HBM в этом случае выступает как кэш и невидим, в двухсокетном сервере будет 8 узлов NUMA. В [280], с учетом полученных там данных о производительности, сформулированы общие рекомендации по оптимальному выбору различных конфигураций NUMA.

В конце обсуждения различных режимов работы с памятью HBM следует отметить, что в Linux для настроек нужны не только средства numactl, но из-за гетерогенности памяти еще и средства daxctl для перенумерации при работе ядра, о чем было упомянуто выше.

Надо также отметить, что хотя HBM дает однозначные преимущества для производительности использующих ее процессоров, кроме повышения их стоимости HBM приводит и к повышению TDP интегрировавших HBM процессоров. Как отмечено в [283], из-за более высокого энергопотребления в 56-ядерном Xeon Max 9480 частота ядер (1.9–3.5 ГГц) чуть ниже, чем в 56-ядерном Xeon 8480+ (2.0–3.8 ГГц).

Ускорители в Xeon Max. Важным преимуществом именно для Xeon Max является AMX-расширение ISA, поскольку HBM-память может существенно поднять производительность задач ИИ. На такое сочетание HBM и средств AMX, поддерживающих актуальные для ИИ матричные операции с данными в форматах BF16 и FP16 особое внимание обращено, например, в [14, 284].

Понятно, что часто производимые расчеты для ИИ обычно эффективнее производить на GPU. Но не для самых сложных вычислений, особенно для выводов ИИ, это может быть и быстрее на x86, так как нет задержек передачи данных на GPU. Анализ, где подобная работа на одном сервере или в Hadoop кластере из них может оказаться эффективнее, безусловно требует дополнительных исследований. Далее в разделе 4 приводятся имеющиеся данные о производительности рассматриваемых в обзоре процессоров Xeon для задач ИИ.

Из других ускорителей в Xeon Max имеется DSA (см. о нем выше в разделе 3.1.3); поддерживается, естественно, и AiA.

3.2.1. *Вычислительные системы на базе Xeon Max*

Выше уже была рассмотрена NUMA-конфигурация 2P-сервера на базе Xeon Max в режиме HBM-only. Совместное применение памяти HBM и DDR дает сложные конфигурации NUMA. На рисунке 30 представлена конструкция 2P-сервера, в котором имеется память HBM и DDR. Это

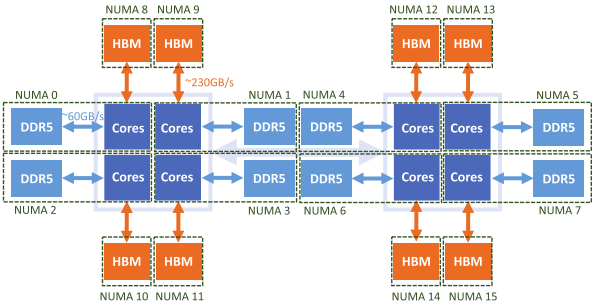


Рисунок 30. Конструкция памяти 2P-сервера с Xeon Max при наличии DDR-памяти (рисунок из [284])

позволяет строить NUMA-конфигурации при сочетании SNC4 с плоским режимом или режимом кэширования. В [280] подробно рассмотрены такие конфигурирования памяти и реализация их в Linux.

После выпуска Intel Xeon Max с HBM-памятью они сразу стали активно применяться в научных исследованиях, и были созданы суперкомпьютеры, использующие эти процессоры в своих узлах. Появившийся с большим запозданием по сравнению с планируемым временем суперкомпьютер Aurora содержит в гетерогенных узлах GPU Intel Data Center Max и процессоры Xeon Max 9470. Также входящий в первую десятку июньского списка TOP500 2024 года суперкомпьютер MareNostrum 5 в своем разделе GPP содержит гомогенные узлы с Xeon Max 9480.

Содержащими только гомогенные узлы с этим же Xeon Max 9480 суперкомпьютерами являются Crossroads (в знаменитой национальной лаборатории Лос-Аламоса) [278], который занимает 27-е место в этом списке TOP500, и Camphor 3[34].

Данные о производительности вычислительных систем с Xeon Max рассматриваются далее в разделе 4.

3.3. Масштабируемые процессоры Xeon 5-го поколения (Xeon EMR)

3.3.1. *Микроархитектура процессоров Xeon EMR и вычислительные системы на их базе*

Микроархитектура Xeon EMR очень близка к изложенному выше в разделе 3.1 для Xeon SPR. Соответственно описание микроархитектурных особенностей Xeon EMR здесь достаточно ограничено. Главным источником информации здесь является публикация разработчиков Intel [285].

Краткое описание Intel [286] дополнительной информации в этом плане содержит очень мало, хотя отдельные подробности можно найти в других документах, например, в руководстве Intel по мониторингу производительности не включающего ядра набора блоков процессора, которые фирма называет «uncore» [287]. В этот набор входят CHA (см. о нем выше в разделе 3.1.3 про микроархитектуру Xeon SPR), блок контроллера питания (PCU, Power Controller Unit), интегрированный контроллер памяти (IMC, Integrated Memory Controller) и другие. В [287] приводится и блок-схема Xeon EMR, которая представлена на рисунке 31.

Из него видно, насколько микроархитектуры Xeon EMR и Xeon SPR похожи (см. также рисунок 26 про Xeon SPR выше).

[34] https://www.iimc.kyoto-u.ac.jp/en/services/comp/supercomputer/system/specification, accessed 13.05.2025.

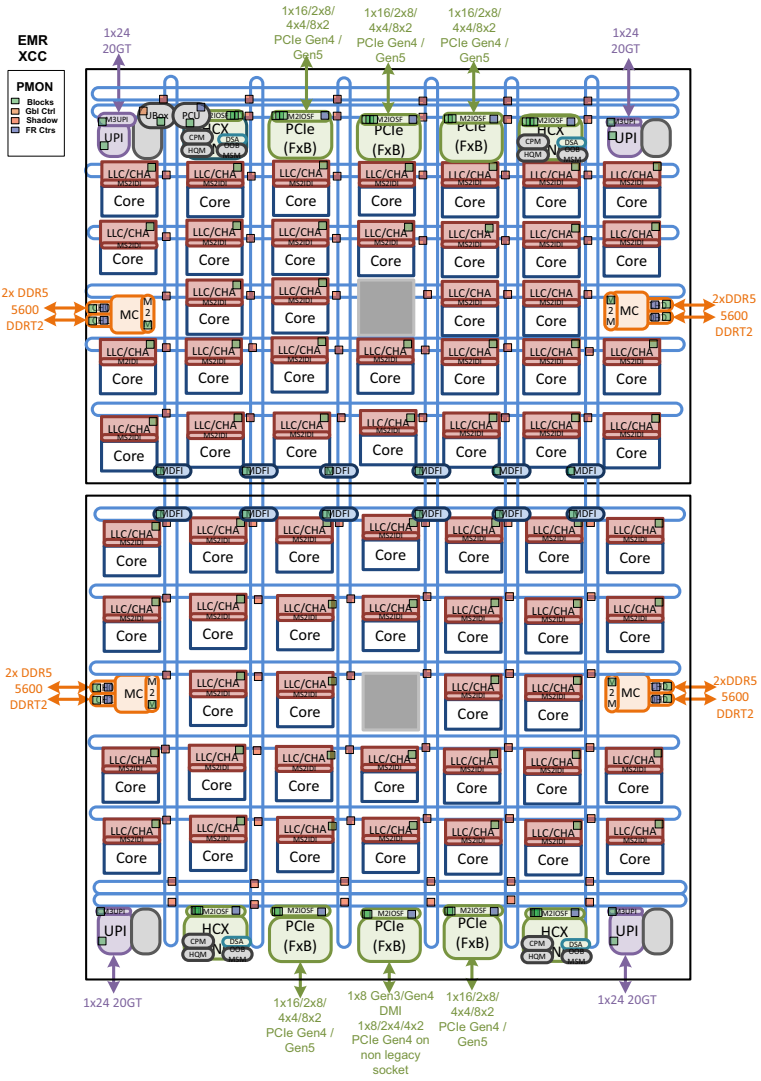


Рисунок 31. Блок-схема Xeon EMR для моделей класса XCC (рисунок из [https://www.intel.com/content/www/us/en/support/articles/000099181/processors/intel-xeon-processors.html 14.05.2025])

Информация об используемой микроархитектуре ядер Xeon EMR приводились скорее сторонними источниками, а в научных публикациях

ссылаются, например, на Википедию [288]. В ядрах Xeon EMR применяется микроархитектура Raptor Cove – небольшое улучшение по сравнению с Golden Cove [289]. Хотя некоторые публикации с данными о Raptor Cove стали появляться [36, 37], но только с точки зрения возможных нарушений безопасности. Поэтому далее мы будем сразу пользоваться данными производителя для конкретных SKU [290].

На самом деле для Xeon EMR имеется 3 группы SKU. Модели с малым числом ядер (Low Core Count, LCC) и со средним числом ядер (Medium Core Count, MCC) являются однокристальными. Модели с экстремальным числом ядер (Extreme Core Count, XCC) являются многокристальными[35]. Именно к ним и относится рисунок 31.

Главными изменениями для этих старших моделях Xeon EMR, судя по данным из приведенных выше ссылок, были переход от применения четырех чиплетов в Xeon SPR к двум и большое увеличение (по сравнению с предыдущим 4-м поколением масштабируемых процессоров Xeon) емкости LLC (см. таблицы 23 и 24). Это было проделано в рамках той же 10 нм технологии Intel 7, которая была оптимизирована, но в первую очередь благодаря проведенной низкоуровневой технологической модернизации (см. [285]). В результате удалось также увеличить максимальное число ядер с 60 до 64 и улучшить энергетические показатели.

Процессоры Xeon SPR содержали четыре плитки (чиплета), а Xeon EMR с конфигурацией XCC теперь содержат две плитки по 32 ядра. Согласно данным [289], другой дизайн в конфигурации MCC использует монолитный кристалл, имея от 20+ до 32 ядер. При этом к LCC относятся модели с 20 ядрами или менее [289]. Это имеет место для специализированных процессоров серии EE (edge-enhanced), Xeon Silver 45xx, которые ориентированы на низкое энергопотребление [286] и в данном обзоре не рассматриваются. Дальнейшее рассмотрение Xeon EMR относится к конфигурации XCC.

Указанное выше в качестве одного из главных улучшений возрастание емкости LLC относится только к моделям высшего класса с конфигурацией XCC и числом ядер от 40 и больше. На самом деле каждая плитка XCC содержит по 33 ядра, но одно отключено для компенсации влияния дефектов во время производства [289]. Число 33 связано с тем, что основа XCC – это сетка ядер 7×5, два из которых заменены контроллерами памяти [291].

В Xeon EMR используется режим полусферы (UMA), аналогичный применяемому в Xeon SPR. Отсутствие в Xeon EMR поддержи имевшейся ранее возможности использования NUMA разными режимами SNC (они рассмотрены выше в начале раздела 3) потенциально может уменьшить

[35] https://www.intel.com/content/www/us/en/support/articles/000099181/processors/intel-xeon-processors.html, accessed 14.05.2025.

эффективность работы с памятью. Но благодаря усовершенствованию производственного процесса удалось даже достигнуть понижения стоимости важных этапов производства. Поскольку кристалла в XCC-конфигурациях теперь всего два, относительно Xeon SPR сильно (до трех) уменьшилось количество используемых мостов межсоединения EMIB. Уменьшились и задержки передачи данных между кристаллами [**285**].

Для Xeon EMR удалось обеспечить очень низкое рабочее напряжение для LLC [**285**], а снижение энергопотребления Intel указывает в качестве одного из ключевых направлений для Emerald Rapids (см., например, [**289**]). Соответственно отмечается улучшение производительности на Вт и снижение TCO [**285**, **286**, **289**, **292**].

В отличие от Xeon SPR, процессоры Xeon EMR поддерживают работу только с одним и двумя сокетами. Важными усовершенствованиями Xeon EMR для межсоединения процессоров по сравнению со своим предшественником является увеличение скорости работы UPI до 20 GT/s против 16 GT/s ранее. Но это требует определенного уточнения. В Xeon Platinum имеется четыре таких канала UPI, в Xeon Gold – три канала, а в Xeon Silver – два канала с UPI, имеющие скорость 16 GT/s [**286**].

Еще одним важным усовершенствованием в Xeon EMR является поддержка 8 каналов для DDR5: с DDR5-5600 (1DPC) и DDR5-4800 (2DPC) для старших моделей. Максимальная емкость памяти на сокет у Xeon EMR составляет 6 ТБ, как и у AMD Genoa. Для потенциального расширения емкости используемой памяти актуальной может оказаться поддержка этими процессорами CXL, что дает возможность подсоединения в будущем еще CXL-памяти. В таблицах 23 и 24 приведена и другая информация о характеристиках Xeon EMR, в том числе о тактовых частотах, в сопоставлении с данными для EPYC 9004. Огромное количество дополнительной информации о не изменившихся относительно Xeon SPR характеристиках (например, о поддержке гиперпоточности по 2 нити на ядро) здесь не приводится.

Серверы на базе Xeon EMR в техническом плане аналогичны серверам с Xeon SPR (как отмечалось выше, у этих процессоров одинаковые разъемы). Главным отличием можно считать отсутствие поддержки у Xeon EMR серверов с 4 или 8 сокетами, что было возможно для Xeon SPR. Естественно, серверы с Xeon EMR предлагают все ведущие производители. В качестве примера укажем на продукцию известной в мире суперкомпьютеров фирмы Lenovo [**293**]. В июньском списке TOP500 2024 года нет суперкомпьютеров с Xeon EMR, а в ноябрьском списке имеются только две системы с гетерогенными узлами, где с 32-ядерными Xeon Gold 6548Y применяются GPU Nvidia H100; более мощная из них, итальянский суперкомпьютер PITAGORA, занимает в TOP500 44-е место.

Имеются многочисленные данные Intel, указывающие на ориентацию Xeon SPR и Xeon EMR на самые разные области применения. Возможность использования в них акселераторов, которые могут применяться еще и в специальном режиме – не с единовременной полной оплатой, а с оплатой зависимости от времени использования, дает дополнительные возможности узкой ориентации конкретных SKU на эффективное применение в конкретной области.

Надо также иметь в виду разделение возможностей разных процессоров на уровне брендов от Platinum до Bronze. В [289] отмечается, что в Xeon SPR имеется 52 варианта SKU, в Xeon EMR – 32, и Intel, вероятно, имеет SKU для каждого типа рабочей нагрузки. Там также отмечается, что стек SKU в EPYC 9004 существенно меньше и более простой для понимания.

В кратком описании Intel [286] кроме направленности Xeon EMR на низкий показатель TCO также указан список его областей применения, начиная от ИИ и баз данных, и заканчивая HPC. В [286] вообще в подзаголовке указано, что это процессор, разработанный для ИИ. Понятно, что здесь успех Xeon EMR может базироваться на сочетании применения AMX с использованием более высокопроизводительного варианта DDR5 и улучшенной энергоэффективности.

Поскольку возможное число ядер в Intel Xeon EMR по-прежнему сильно отстает от доступного в AMD EPYC 9004, и имеется также отставание в числе каналов памяти и соответственно в ее пропускной способности, в вычислительно сложных рабочих нагрузках, включая распараллеливание, для старших моделей Xeon EMR часто предполагается отставание от EPYC 9004 по производительности (см., например, [289]).

Хотя некоторые данные о производительности Xeon EMR иногда вроде как показывают ее уменьшение относительно Xeon SPR [289], всегда нужно аккуратно уточнять смысл полученного. Несомненно, что Xeon EMR вплоть до появления Xeon 6 являлись лучшими серверными процессорами Intel для традиционных двухсокетных систем. Дополнительной иллюстрацией этого являются и стоимостные показатели (например, в таблице 23 видна уменьшенная цена старшей модели Xeon EMR по сравнению с Xeon SPR). Некоторые грубые средние оценки отношения стоимость/производительность для разных SKU Xeon EMR приведены в [291].

Данные о производительности систем с Xeon EMR рассматриваются далее в следующем разделе.

4. Данные о производительности Xeon SPR, Xeon Max и Xeon EMR

Мы объединяем рассмотрение производительности нескольких семейств процессоров Xeon в один общий раздел, поскольку они отнесены

к общему условно 4-му поколению x86, обладают достаточно близкими спецификациями и могут заменяться в рамках одного сервера, что обеспечивается благодаря применению одного и того же разъема (Socket 4677). Такое объединение в рамках одного документа применяла, например, и Fujitsu для информации о производительности своих серверов с этими процессорами (см., например, [**166**]).

Этому соответствует и суммарный объем известных нам структурированных данных о производительности всех этих процессоров (соответствующих серверов), сопоставимый с имеющимся для EPYC 9004. Исключением являются задачи ИИ, для которых имеется много публикаций, а Intel предоставляет и специальные руководства по оптимизации – например, для PyTorch [**294**] или для своего расширения TensorFlow, использующего Python [**295**]. Поэтому производительность в ИИ рассматривается в отдельном подразделе 4.2.

Необходимо также указать на большое количество данных о производительности Xeon SPR и Xeon EMR, включая сопоставления их производительности, в неструктурированной форме [**296**, **297**], которые здесь не рассматриваются.

В основном доступные и анализируемые данные о производительности Xeon относятся к старшим моделям с большим числом ядер. В подразделе 4.3 рассматривается производительность процессоров Xeon SPR и Xeon EMR в сопоставлении с GPGPU, а также моделей Xeon среднего класса с аналогичными процессорами EPYC 9004.

Мы также выделяем данные о производительности Xeon Max в отдельный подраздел 4.4. Это связано с тем, что поддержка двух возможных уровней памяти, HBM и DDR, является уникальной, и вызвала подъем числа научных публикаций на эту тему. Поддержка HBM была прекращена в последующем семействе Xeon EMR, и отсутствует и в новейших Xeon 6 – так что развитие по линии Xeon SPR-Xeon EMR-Xeon 6 является основным вектором усовершенствования серверных процессоров Intel.

4.1. Производительность Xeon SPR и Xeon EMR в тестах и приложениях

Мы далее сопоставляем производительность Xeon SPR с предыдущим поколением Xeon ICL достаточно ограничено, поскольку преимущество в этом плане Xeon SPR очевидно. Очень большое количество разнообразной информации для таких сравнений имеется на сайте Intel [**296**]. Мы проиллюстрируем все это одним имеющим общий характер рисунком 32. Там приведены данные о производительности 2P-серверов с 56-ядерными Xeon 8480+, Xeon Max 9480 и 40-ядерными Xeon 8380 в приложениях из самых разных областей применения. Два примера там представлены для 60-ядерных Xeon 8490H (см. подробнее в [**298**]).

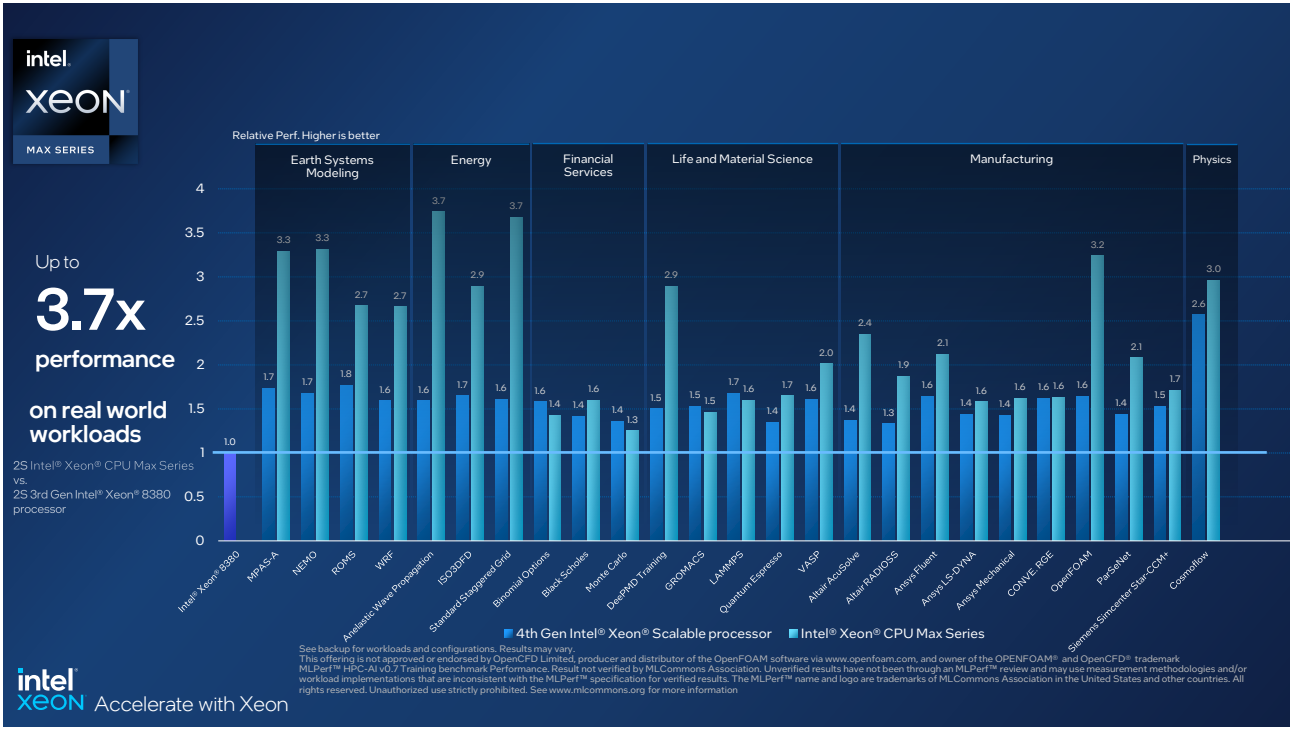


Рисунок 32. Сопоставление производительности 2P-серверов с Xeon 8380, Xeon 8480+ и Xeon Max 9480 (рисунок из [298])

Производительность в различных тестах SPEC. В таблице 27 приведены данные о производительности Xeon SPR, Xeon EMR и EPYC 9004 в тестах SPECcpu 2017 с плавающей запятой. Анализ данных этой таблицы может дать ряд полезных сведений. Понятно, что часто требуются дополнительные уточняющие и поясняющие данные научных публикаций, но учитывая большую интегральность этих показателей и массовость попыток их улучшения разными производителями серверов некоторые выводы полезно сделать сразу здесь.

Аналогичные данные для 3-го поколения масштабируемых процессоров Xeon и EPYC Zen 3 (7003) уже приводились выше в таблице 1. Сравнивая эти таблицы, можно увидеть большой прогресс в производительности Xeon SPR по сравнению с предыдущим поколением Xeon не только благодаря увеличению числа ядер, но и при одинаковом числе ядер (сравнив, например, 32-ядерные модели Xeon 8362 и Xeon 8462Y+).

Модели с большим числом ядер обычно превосходят по производительности модели с меньшим числом ядер, что дает преимущества EPYC 9004. При равном числе процессорных ядер (32) показатели EPYC 9374F выше, чем у Xeon SPR 8462Y+ (только в base-варианте fp_speed EPYC чуть отстал).

Xeon EMR дает существенное повышение производительности по сравнению с Xeon SPR не только благодаря повышению числа ядер в старших моделях, но и при одинаковом числе ядер (например, можно сравнить данные для 32-ядерных и 60-ядерных моделей Xeon в таблице 27). Однако по возможному числу ядер старшие модели Xeon EMR по-прежнему сильно отстают от EPYC 9004, что отражается более высокими данными EPYC об их производительности в таблице 27.

Проиллюстрируем производительность в SPECcpu сначала сопоставлением процессоров Xeon EMR и EPYC 9004 при одинаковом числе ядер. 64-ядерные процессоры Xeon 8592+ чуточку отстают от EPYC 9534 по производительности, имея при этом несколько более высокую цену. 48-ядерные процессоры Xeon 8558P на 2P-сервере в тесте fp_rate чуть опережают EPYC 9474F по производительности, хотя уступают им в 1P-конфигурациях. Но разница в производительности не так велика, и следует обращать больше внимания на TDP и стоимость (и соответственно на TCO). Цена Xeon 8558P чуточку ниже, но после добавления стоимости чипсета становится выше. При сравнении 32-ядерных процессоров, Xeon EMR 8562Y+ и EPYC 9374F, Xeon EMR заметно отстает в тестах fp_rate (возможно, более интересных для облачных технологий), и близок по производительности к EPYC 9374F в тесте fp_speed, где достигаемый уровень распараллеливания может быть низким. При этом цена Xeon 8562Y+ существенно выше.

Но у EPYC 9004 есть модели с большим числом ядер, и они опережают Xeon по производительности гораздо больше.

Индивидуальная оптимизация отдельных составляющих тестов SPECcpu 2017 fp_speed (что отражается в пиковых показателях) сильнее влияет на производительность в EPYC, чем в Xeon. Масштабирование производительности с ростом числа ядер при переходе от одной модели процессора к другой в тестах fp_speed не такое сильное: увеличение числа ядер в два раза не дает близкий к двукратному рост производительности (хотя у процессоров с малым числом ядер их частота обычно несколько выше). Зато в тестах fp_rate при переходе от одного к двум процессорам наблюдается высокое масштабирование производительности. Это естественно соответствует типам самих этих тестов.

В обоих типах тестов SPECcpu 2017 обнаруживается, что вовсе не обязательно небольшое увеличение числа используемых процессорных ядер при переходе к более дорогой модели процессора той же архитектуры приводит к росту достигаемой производительности. Согласно данным таблицы 27, 2P-сервер с Xeon 8580, имеющим на 4 ядра больше Xeon 8570 при чуть более низкой частоте, получил более низкие данные производительности. Да, эти результаты зависят от уровня оптимизации производящего тесты, но в 1P-сервере Xeon 8580 показывал более высокую производительность.

Сказать, что при одинаковом числе ядер в Xeon 5-го поколения и EPYC 9004 производительности в этих тестах близки, на основании этих данных нельзя. Мы еще отдельно проведем далее небольшое сопоставление процессоров Xeon и EPYC среднего класса, которые могут активно использоваться и в HPC [**100**].

Хотя в таблице 27 приведены данные тестов SPECcpu 2017 и для Xeon Max, их производительность будет рассмотрена далее в отдельном разделе 4.4.

Тесты SPEC CPU 2017 с целочисленной арифметикой практически не представляют интереса для данного обзора, особенно для HPC. В SPECint_speed 2017, где распараллеливание слабо влияет на результат, Xeon 8592+ на 1% опередил EPYC 9684X, и на 5% – EPYC 9654. Но в SPECint_rate 2017 старшие модели EPYC 9004, естественно, сильно впереди (все это относится к максимальным достигнутым на 5.02.2024 результатам).

Таблица 27. Сопоставление производительности старших моделей Xeon SPR, Xeon EMR, Xeon Max и EPYC 9004 в тестах SPECcpu 2017 с плавающей запятой

Модели процессора	Число CPUs × ядер	Частота, ГГц	Цена CPU[1]	SPECcpu 2017 fp_speed (base/peak)[2]	SPECcpu 2017 fp_rate (base/peak)[2]
EPYC 9754	128 2×128	2.25–3.1	$11900	316/318[4] 430/448[3]	733/799[4] 1460/1590[3]
EPYC 9654	96 2×96	2.4–3.7	$11805	324/327[6] 449/468[3]	746/784[7] 1480/1570[5]
EPYC 9684X	1×96 2×96	2.55–3.7	$14756	357/358[4] 476/495[3]	820/854[4] 1630/1700[3]
EPYC 9534	1×64 2×64		$8803	298/305[4] 427/442[3]	610/658[4] 1220/1310[3]
EPYC 9474F	1×48 2×48	3.6–4.1	$6780	277/290[4] 405/430[3]	574/575[4] 1150/1150[3,5]
EPYC 9454	1×48 2×48	2.75–3.8	$5225	264/278[9] 388/423[3]	555/557[7] 1100/1110[8]
EPYC 9374F	1×32 2×32	3.85–4.3	$4850	256/272[4] 361/385[3]	481/485[4] 964/968[3]
Xeon 8490H	1×60 2×60	1.9–3.5	$17000	255/—[10] 378/378[10]	508/—[10] 1040/1100[10]
Xeon 8480+	1×56 2×56	2–3.8	$10710	242/242[13] 371/371[10]	465/337[13] 1020/1080[10]
Xeon 8462Y+	2×32	2.8–4.1	–	363/363[1,11,12]	817/853[10]
Xeon 8592+	1×64 2×64	1.9–3.9	$11600	286/—[14] 428/428[10]	619/646[15] 1260/1300[10]
Xeon 8580	1×60 2×60	2.0–4.0	$10710	286/—[10] 419/419[16]	570/—[10] 1160/1190[16]
Xeon 8570	1×56 2×56	2.1–4.0	$9595	279/—[14] 423/423[10]	544/—[14] 1220/1260[8]
Xeon 8558P	1×48 2×48	2.7–4.0	$6759	275/—[14] 412/412[17]	526/—[14] 1140/1170[17]
Xeon 8568Y+	1×48 2×48	2.3–4.0	$6497	275/—[14] 413/413[17]	526/—[14] 1170/1210[10]
Xeon 8558	1×48 2×48	2.1–4.0	$4650	263/—[14] 412/412[17]	512/—[14] 1090/1120[10]
Xeon 8562Y+	1×32 2×32	2.8–4.1	$5945	261/—[14] 388/388[10]	414/—[14] 908/941[17]
Xeon 8592V	1×64 2×64	2–3.9	$10995	273/—[14] 409/409[16]	542/—[13] 1200/1250[10]
Xeon Max 9480	2×56	1.9–3.5	$12980	348/349[16]	1140/1160[18]
Xeon Max 9470	2×52	2–3.5	$11590	347/349[18]	1100/1120[18]

[1] Данные о цене 1 CPU на 13.05.2024 для AMD – из [49], для Intel – из [https://ark.intel.com/content/www/us/en/ark.html#@PanelLabel595 23.02.2025]
[2] Данные с максимальной представленной пиковой величиной на 28.08.2024. Отправители результатов теста и серверы, на которых получены эти результаты:
[3] ASUS RS720A-E12-RS12;
[4] ASUS RS520A-E12-RS12U;
[5] xFusion FusionServer 2258 V7;
[6] ASUS RS520A-E12(K14PA-U24);
[7] xFusion FusionServer 1158H V7;
[8] Lenovo ThinkSystem SR675 V3;
[9] Lenovo ThinkSystem SR655 V3;
[10] ASUS ESC4000-E11;
[11] xFusion FusionServer 2288H V7;
[12] ASUS RS720-E11-RS12U(Z13PP-D32);
[13] Supermicro UP SuperServer SYS-521C-NR (X13SEDW-F);
[14] Lenovo ThinkSystem SD530 V3;
[15] IEIT meta brain i24G7;
[16] Nettrix R620 G50;
[17] ASUS RS720-E11-RS12U;
[18] Dell PowerEdge C6620.

Примечание. Для уточнения данных о применявшихся серверах в круглых скобках иногда указана использовавшаяся в тестах материнская плата.

Среди данных разных тестов SPEC для задач HPC, безусловно, тесты SPEChpc 2021 относятся к наиболее актуальным. Очень интересна работа [**299**], в которой детально исследованы результаты выполнения тестов SPEChpc 2021 (в tiny-варианте) для 2P-серверов с 36-ядерными Xeon ICL 8360Y и с 52-ядерными Xeon SPR 8470, и для кластеров с этими серверами (в small-варианте SPEChpc 2021). Хотя здесь изучены варианты с распараллеливанием только с MPI, эта статья дает информацию, выходящую за пределы конкретных моделей процессоров на уровень Xeon ICL и Xeon SPR вообще и представляющую интерес для проведения анализа SPEChpc на более новых процессорах x86 или других современных процессорах.

В [**299**] проведено разбиение отдельных тестов из SPEChpc на группы связанных памятью и вычислительно-интенсивных, и детально исследовано масштабирование производительности в зависимости от числа MPI-процессов при оптимальной для пропускной способности памяти настройке ccNUMA (SNC4 для Xeon SPR). Кроме того, при этом детально исследуется энергопотребление процессорами и DRAM, и рассеивание мощности, и оценивается энергоэффективность с применением EDP (Energy Delay Product) (про EDP см., например, [**300**]).

В [**299**] было найдено, что ускорение в вычислительно-интенсивных тестах из SPEChpc при переходе от Xeon ICL к Xeon SPR было меньше или практически не превышало увеличения числа ядер в Xeon SPR по сравнению с Xeon ICL, в то время как связанные памятью тесты получили большее ускорение (до двух раз для теста weather) – см. таблицу 28.

Таблица 28. Ускорение в сервере с Xeon SPR по сравнению с Xeon EMR в SPEChpc 2021 (tiny) при MPI-распараллеливании

(a) Вычислительно-интенсивные тесты

	lbm	soma	sweep	sph-exa
Ускорение	1.21	1.35	1.39	1.48

(b) Связанные памятью тесты

	weather	tealeaf	cloverleaf	pot3d	pgmgfv
Ускорение	2.03	1.66	1.57	1.63	1.65

Что касается вопросов энергии, то в [**299**] были определены «горячие» и «холодные» тесты из состава SPEChpc 2021, имеющие с высокое и низкое рассеивание мощности на процессоре. К первым относятся вычислительно-интенсивные тесты, в которых энергопотребление близко к TDP, большая часть мощности потребляется вычислительными блоками и кэш-памятью, а в DRAM используется маленькая часть мощности. Связанные памятью тесты, наоборот, относятся к «холодным», где достигается высокая мощность DRAM.

Было показано, что память DDR5 в Xeon SPR является гораздо более энергоэффективной и меньше влияет на общую потребляемую сервером мощность, чем DDR4 в Xeon ICL, хотя общая емкость памяти в серверах с Xeon SPR была в 4 раза больше. Однако вклад от DRAM в общую потребляемую мощность невелик. Было найдено, что в режиме простоя Xeon ICL и Xeon SPR имеют высокий уровень мощности (50% от TDP для Xeon SPR), в то время как в Xeon Sandy Bridge, например, было только 20% от TDP. В [**299**] сделан вывод, что для вычислительно-интенсивных тестов Xeon SPR менее эффективен, поскольку рост потребляемой мощности не компенсируется ростом производительности.

Результаты [**299**], относящиеся к кластеру, связаны с использованием обменов данных между узлами и здесь не рассматриваются.

В таблице 29 приведены наилучшие «официальные» результаты SPEChpc 2021 для топ-моделей Xeon SPR, Xeon EMR и EPYC 9004. Жирным шрифтом здесь помечены максимальные достигнутые на традиционных 2P-серверах базовые результаты для каждой модели процессоров. Эти данные относится к производительности серверов, а полученные в кластерах здесь не приводятся, поскольку они в меньшей степени отражают производительность самих процессоров.

Таблица 29. Данные о производительности серверов со старшими моделями EPYC 9004 и Intel Xeon в тестах SPEChpc 2021

Модели	Число ЦП	Tiny base/peak	Small base/peak
EPYC 9654	1 2	6.99/6.99 **13.9**/14.2	0.735/0.735 **1.45**/1.45
EPYC 9684X	2	**16.0**/16.0	**1.55**/1.55
EPYC 9754	1 2	7.32/ **16.4**/	0.823/ **1.59**/
Xeon 8480+	2	**7.98**/8.35	**0.945**/0.949
Xeon 8490H	2 4	**9.00**/ 17.2/17.6	**1.00**/ 1.88/1.89
Xeon 8592+	1 2	4.88/ **10.8**/	0.553/ **1.15**/

Данные из http://spec.org/hpc2021/results/ на 12.09.2024

Из таблицы видно, что производительность была выше при большем числе использовавшихся ядер (в процессорах применялись все ядра). Но наивысшие результаты получены на четырехпроцессорном сервере с 60-ядерными Xeon SPR 8490H, хотя в нем немного меньше ядер, чем в 2P-сервере со 128-ядерными EPYC Bergamo 9754. В Bergamo по сравнению с Genoa меньше емкость кэша L3 на ядро, а его емкость достаточно сильно влияет на производительность, что видно и из сопоставления результатов для 96-ядерных EPYC 9684X с 3D V-cache и EPYC 9654. Успех четырехпроцессорного сервера связан, вероятно, и с тем, что производительность лучше масштабируется при использовании нескольких сокетов. Среди 2P-серверов содержащие EPYC 9004 имеют большую производительность, чем серверы с Xeon.

К тестам SPEChpc 2021 в определенном смысле можно отнести работу [**301**], в которой исследованы различные данные о производительности входящего в SPEChpc мини-приложения CloverLeaf на 2P-сервере с Xeon 8480+ и сопоставлены с Xeon ICL.

Следующими по актуальности для задач HPC можно было бы считать тесты SPEComp 2012 и SPECmpi 2007. Данные SPECmpi для Xeon SPR или Xeon EMR на момент написания обзора отсутствовали, а данные SPEComp представлены выше в таблице 7. 2P-серверы с топ-моделями Xeon SPR или Xeon EMR сильно уступали по производительности топ-моделям EPYC 9004, что достаточно естественно вследствие гораздо меньшего числа процессорных ядер, чем в моделях EPYC.

Можно также отметить слабое увеличение производительности в 2P-сервере с 64-ядерными Xeon 8592+ по сравнению с 56-ядерными Xeon 8480+ или с 60-ядерными Xeon 8490H (по сравнению с последними рост производительности составил около одного процента). Однако производительность 2P-сервера с Xeon 8592+ близка к производительности имеющего столько же (128) ядер 1P-сервера с EPYC 9752 Bergamo, что может быть связано с уменьшенной емкостью кэша L3 на ядро в Bergamo и повышенной пропускной способностью памяти при работе с 2P-сервером.

Самые высокие показатели SPEComp были получены на сервере с Xeon 8490H, но это имеет место для 8-процессорного сервера. Данных, позволяющих сопоставить производительность Xeon и EPYC 9004 на серверах с одинаковым числом сокетов и ядер, на конец 2024 года не имелось.

Из данных других тестов SPEC для процессоров Xeon по сравнению с EPYC 9004 укажем данные о производительности для серверов Java, тестов SPECjbb2015, которые были представлены выше в таблице 9. Эти данные показывают увеличение производительности в традиционных 2P-серверах при переходе на процессоры с большим числом ядер как в семействе Xeon SPR, так и в семействе Xeon EMR. Данные таблицы в тесте composite для critical_JOPS для 2P-серверов с Xeon 8490H по сравнению с Xeon 8480+ увеличения производительности не показывают, но позднее южнокорейская фирма KTNF на своем сервере получила более высокий показатель, 335807 единиц.

Масштабирование производительности с числом процессоров Xeon SPR в сервере до восьми для этих тестов можно считать вполне удовлетворительным, кроме варианта composite, где масштабирование плохое уже при переходе от двух к четырем процессорам.

Сопоставление производительности при одинаковом числе ядер у 56-ядерных 8480 и 8570, 60-ядерных Xeon 8490H и Xeon 8580 не показывает ее четкого роста при переходе от Xeon SPR к Xeon EMR. Правда, это может быть связано просто с маленьким количеством представленных пока результатов (количестве попыток оптимизации) для Xeon 8570 и 8580. Сервер с топ-моделью Xeon EMR 8592+ существенно опередил по производительности сервер с топ-моделью Xeon SPR 8490H.

Но серверы со старшими моделями масштабируемых процессоров Xeon 4-го и 5-го поколений сильно отстают по производительности в этих тестах от 4-го поколения EPYC 9004, и в некоторых случаях 2P-серверы с Xeon отстают от 1P-серверов с EPYC 9004, а четырехпроцессорные серверы с Xeon – от 2P-серверов с EPYC 9004, хотя при этих сравнениях число процессорных ядер в серверах с Xeon было больше, чем в серверах с EPYC.

Что касается тестов производительности на Ватт, SPECpower_ssj 2008, то процессоры Xeon SPR и Xeon EMR превосходят по этому показатели ARM-процессоры Ampere Altra (см. таблицу 10 выше), и Xeon EMR продемонстрировали улучшение в этом плане по сравнению с Xeon SPR. Однако по данным этой таблицы Xeon существенно уступают EPYC 9004.

Кроме данных сайта этого теста, имеются интересные данные SPECpower_ssj 2008 для 2P-серверов Fujitsu с 60-ядерными Xeon 8490H и 64-ядерными Xeon 8592+ [**166**], с 32-ядерными Xeon 6438Y+ и Xeon 6538Y+ [**302**], а также с 32-ядерными Xeon 6428N [**303**] при различных использованных параметрах.

Приведенные выше данные говорят о преимуществах по производительности старших моделей EPYC 9004 по сравнению с Xeon SPR и EMR. Пожалуй, в качестве единственного исключения среди тестов SPEC (если не считать SPECint_speed 2017), в котором наблюдались успехи Xeon SPR или EMR по сравнению с EPYC 9004, здесь можно указать тесты виртуализации SPECvirt_sc2013 (см. таблицу 11 выше). В них 2P-сервер с Xeon EMR 8592+ сумел опередить 2P-сервер с EPYC 9654 (серверы с Xeon SPR от него отстали). Эти тесты, с одной стороны, интересны для массовых ЦОД вследствие интеграции широко распространенных рабочих нагрузок, а с другой стороны в меньшей степени относятся к производительности собственно процессоров, и полученных результатов для сравнения здесь мало.

Здесь можно отметить также резкое удешевление старших моделей 5-го поколения масштабируемых процессоров Xeon EMR по сравнению с 4-м поколением Xeon SPR. Даже требующая жидкостного охлаждения самая старшая 64-ядерная модель Xeon 8593Q стоит $12400 (цена на 15.03.2025). Но сравнение производительности с учетом цен по сравнению со старшими моделями EPYC 9004 (см. также обсуждение выше в разделе 2.4) преимущества EPYC из-за этого кардинально не изменяет.

Пропускная способность памяти. В качестве общего введения в анализ количественных показателей пропускной способности памяти при работе с Xeon SPR и Xeon EMR можно использовать данные тестов stream для 2P-серверов Fujitsu [**304**]. Там для stream triad приведены зависимости относительной величины пропускной способности для серверов с 56-ядерными Xeon 8570 и Xeon 8480+, 32-ядерными Xeon 6548N и Xeon 6430 с отключенным SMT от разных типов DIMM по числу рангов на канал памяти, ширины столбцов DRAM, емкости DIMM, 1DPC или 2DPC-конфигураций и числа DIMM на сокет. В качественном плане с точки зрения увеличения/уменьшения пропускной способности влияние большинства из этих параметров предсказуемо.

Данные о пропускной способности в stream triad для широкого набора разных моделей Xeon SPR представлены в таблице 30 (данные для 2P-серверов Fujitsu). Все использованные в тестах серверы работали с памятью на 4800 MT/s, и пиковая пропускная способность памяти одного процессора у всех этих моделей составляет 307 ГБ/с.

Таблица 30. Пропускная способность памяти в 2P-серверах с разными моделями Xeon SPR

Модель	Число ядер	Частота, ГГц	Пропускная способность, Гбайт/с
Xeon 8490H	60	1.90	523
Xeon 8480+	56	2.00	518
Xeon 8470Q	52	2.10	492
Xeon 8470N	52	1.70	487
Xeon 8470	52	2.00	511
Xeon 8468V	48	2.40	490
Xeon 8468	48	2.10	485
Xeon 8460Y+	40	2.00	469
Xeon 6458Q	32	3.10	444
Xeon 6454S	32	2.20	445

Данные из [**303**].

Мы видим здесь, что переход от более младшей модели с меньшим числом ядер к более старшей модели демонстрирует увеличение пропускной способности (при отборе наиболее высокопроизводительных моделей при одинаковом числе ядер). Кроме того, соответствующее смене модели процессора повышение частоты ядер при их одинаковом числе также обычно повышает пропускную способность.

Что касается Xeon EMR, то для старшей 64-ядерной модели Xeon 8592+ сервер Fujitsu в 2P-конфигурации показал заметно более высокую пропускную способность, 577 ГБ/с. Такое повышение пропускной способности по сравнению с Xeon SPR, естественно, приводит и к росту производительности для других связанных памятью тестов. Так, для двумерного БПФ в тесте FFTW 2P-сервер с Xeon 8592+ дает ускорение в среднем на 22% (максимум – на 68%) относительно 2P-сервера с Xeon 8480+ [**305**].

Можно отметить, что достигаемая пропускная способность старших моделей и Xeon SPR, и Xeon EMR сильно уступает соответствующим данным для EPYC 9654, EPYC 9684X и EPYC 9754 (см. выше раздел 2.4.2), где достигается свыше 750 ГБ/с. Но если из этих чисел посчитать пропускную способность на одно ядро, то у 60-ядерного Xeon 8490H и 64-ядерного Xeon 8592+ она получается выше, чем у 96-ядерных процессоров EPYC.

Зависимость пропускной способности памяти в 2P-сервере с Xeon 8480+ в тесте stream от числа используемых ядер (на оси абсцисс, в логарифмическом виде) представлена на рисунке 33.

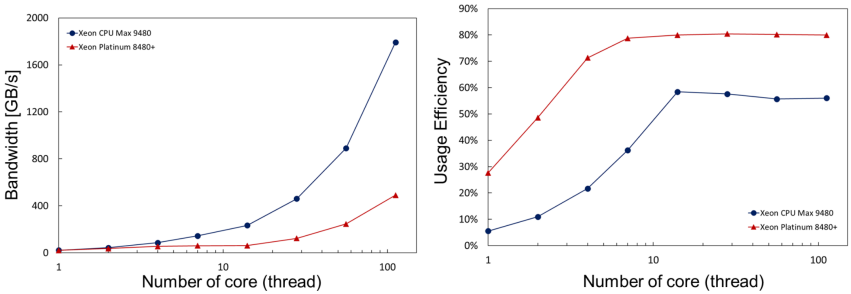


Рисунок 33. Масштабирование достигаемой пропускной способности памяти в 2P-серверах с Xeon Max 9480 и Xeon 8480+ от числа ядер (рисунок из [**283**])

Он показывает, что масштабирование пропускной способности в определенной степени имеет место вплоть до максимального числа имеющихся ядер. Хотя эффективность использования дополнительных ядер после определенного порога падает.

Отдельный интерес представляют данные о пропускной способности памяти в 4- и 8-процессорных серверах Fujitsu с Xeon SPR. Для

4-процессорного сервера с 60-ядерными Xeon 8490H в stream triad получено 805 ГБ/с против 1600 ГБ/с для 8-процессорного сервера. Для 4-процессорного сервера с 40-ядерными Xeon 8460H достигнутая пропускная способность составила 817 ГБ/с против 1593 ГБ/с для 8-процессорного сервера. Эти данные представлены в [**63**] и [**306**], и демонстрируют ожидаемо хорошее масштабирование с ростом числа используемых процессоров.

Производительность при умножении плотных матриц и в тесте HPL. К сожалению, автору неизвестны данные о производительности 2P-серверов с топ-моделями Xeon SPR и Xeon EMR (а не их отдельных ядер) с применением для умножения плотных матриц в формате FP64 средств DGEMM из MKL. Однако для 48-ядерного Xeon 8468, имеющего пиковую производительность около 3.2 TFLOPS, имеются данные для японского суперкомпьютера Pegasus, который использует эти процессоры в узлах совместно с GPU Nvidia H100 [**307**]. Там с применением oneMKL была достигнута производительность около 3 TFLOPS (93% от пиковой), при этом включение турбо-режима там не давало существенного ускорения.

Для FP32 имеются данные о соответствующих временах выполнения на 2P-сервере с Xeon 8480+ при использовании средств MKL [**4**].

Здесь необходимо также отметить работы с применением AMX, которые относятся не только к задачам ИИ, но и вообще к расчетам со смешанной точностью (см., например, [**308**]). AMX используется и в oneMKL. Intel в [**309**] приводит данные о производительности 2P-сервера с 56-ядерными Xeon 8480+ при умножении квадратных матриц разных размеров с применением AMX – в форматах FP32 и BF16. Эта информация представлена на рисунке 34.

Имеются еще интересные данные о сопоставлении производительности при использовании AMX для GEMM и GEMV в форматах BF16/FP16 в 2P-сервере с 40-ядерными Xeon 8460H по сравнению с GPU Nvidia P100 и A100, причем в GEMM при разных размерах матриц два процессора Xeon чаще опережали P100 [**310**].

Для этих форматов производительность 2P-сервера с Xeon 8480+ с использованием других программных реализаций GEMM представлена в [**311**], где она сопоставлена, в частности, с данными для 2P-сервера с ARM-процессорами AWS Graviton 3, содержащими по 64 ядра Neoverse V1. Естественно, достигаемая Xeon 8480+ производительность оказалась сильно выше. Соответствующие данные рассматриваюися далее в разделе 4.2 про производительность Xeon в задачах ИИ.

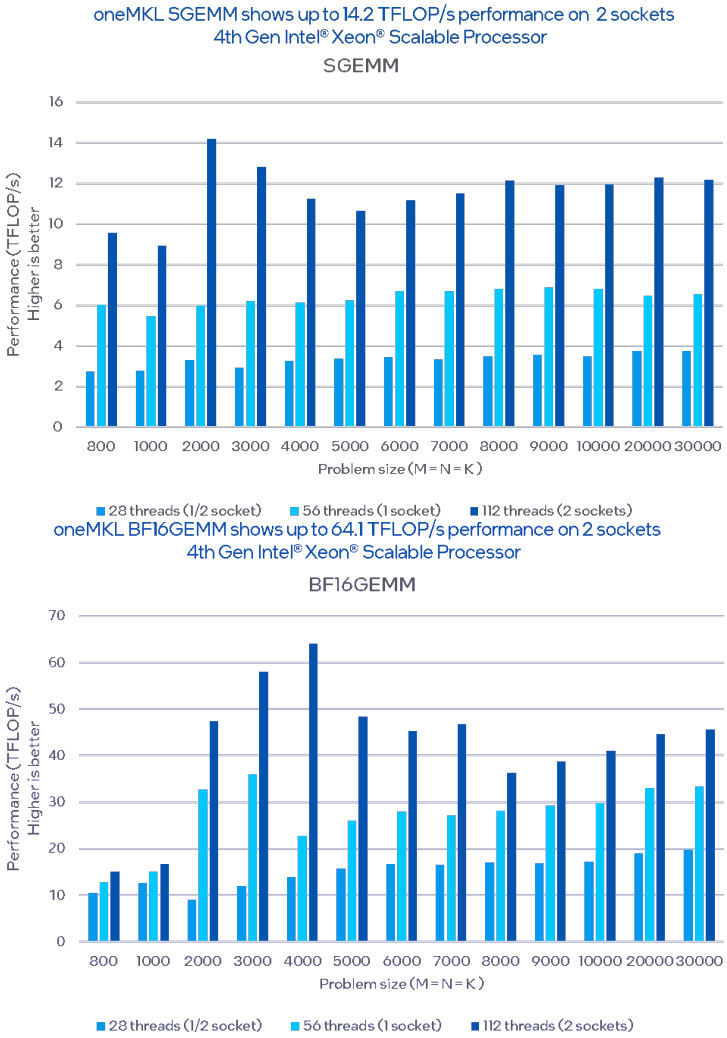


Рисунок 34. Производительность 2P-сервера с Xeon 8480+ при умножении матриц в форматах FP32 и BF16 (рисунок из [**309**])

При этом работы с информацией о производительности GEMM половинной точности, на которых мы ссылались выше, ориентировались на задачи ИИ и, в частности, большие языковые модели.

Данные о производительности Xeon SPR и Xeon EMR в тесте HPL предоставлялись производителями серверов с этими процессорами. В таблице 31 приведены данные о производительности серверов Fujitsu PRIMERGY с Xeon SPR и Xeon EMR в HPL для RX2540 M7 [**166**] и для CX2550 M7 [**303**].

Таблица 31. Производительность 2P-серверов с Xeon SPR в тесте HPL

Модель	Число ядер	Частота, ГГц	Производительность, GFLOPS	
			Данные [**303**]	Данные [**166**]
Xeon 8490H	60	1.9	7584	7386
Xeon 8480+	56	2.0	7293	7388
Xeon 8470Q	52	2.1	7051	
Xeon 8470N	52	1.70	6264	6105
Xeon 8470	52	2.00	7051	6930
Xeon 8468V	48	2.4	7022	6321
Xeon 8468	48	2.1	6698	6544
Xeon 8458P	44	2.7		6162
Xeon 8460Y+	40	2.00	5538	5421
Xeon 8462Y+	32	2.80		5522
Xeon 6548Q	32	3.10	6160	
Xeon 6454S	32	2.20	4667	4418
Xeon 6428N	32	1.80	4025	3826
Xeon Max 9480	56	1.90	6781	

Данные таблицы 31 дают определенные оценки того, как производительность может меняться с увеличением числа ядер или тактовых частот. Понятно, что представленные в этой таблице данные от Fujitsu могут быть не максимально достижимые показатели, которые зависят, в частности, и от размерности применяемых матриц. Для Xeon 8592+ в [**303**] указана производительность 8542 GFLOPS.

Эти данные о производительности для Xeon 8480+, Xeon 8490H и Xeon 8592+ выше, чем для EPYC 9654, приведенные в таблице 15. Однако максимальная приведенная AMD в [**121**] для 2P-сервера с EPYC 9654 производительность, 8856 GFLOPS, существенно больше, чем у Xeon SPR, и немножко больше, чем у Xeon 8592+. Аналогичные данные в [**121**] для EPYC 9684X и EPYC 9754 (они приведены в разделе 2.4.3) также больше, чем у этих процессоров Intel.

Различие связано в первую очередь с большим числом ядер в данных моделях EPYC, чем у Xeon SPR и Xeon EMR. Определенное отставание EPYC по сравнению с Xeon в количестве операций с плавающей запятой за такт у ядер EPYC, в отличие от умножения матриц, здесь отчасти нивелируется, поскольку в HPL на общее время расчета влияет не только умножение матриц.

В [**312**] для сервера с 56-ядерными Xeon 8480+ приведены полученные японской компанией HPC Systems данные о производительности в HPL в зависимости от размерности матрицы N. Соответствующие данные представлены в таблице 32.

Таблица 32. Производительность в HPL на сервере с Xeon 8480+

N	140000	180000	200000	220000
Производительность, GFLOPS	4414.2	4683.8	4764.9	4819.0

Эти результаты были получены с использованием классического компилятора Intel из oneAPI Base & HPC Toolkit 2022.2.0, а также соответствующей версии oneMKL.

Кроме того, имеются данные о производительности Xeon SPR в HPL на 4- и 8-процессорных серверах от Fujitsu [**63**, **306**], которые представлены в таблице 33.

Таблица 33. Масштабирование производительности HPL с числом процессоров Xeon SPR

Модель	Число ядер	Частота	Производительность, GFLOPS			
			4 процессора		8 процессоров	
			HPL	Пиковая	HPL	Пиковая
Xeon 8490H	60	1.90 ГГц	14505	14505	25061	29184
Xeon 8468H	56	2.10 ГГц	12656	12902	23274	25805
Xeon 8460H	52	2.20 ГГц	11872	11264	21697	22528

Эти данные говорят о хорошей масштабируемости производительности HPL с числом процессоров в сервере и о ее близости к пиковой величине. Здесь надо отметить, что в тестах был включен турбо-режим, а пиковая производительность считается на базовых тактовых частотах.

Хотя традиционные 2P-серверы со старшими моделями Xeon SPR и Xeon EMR по производительности в HPL уступают серверам со старшими моделями EPYC 9004, 4- и 8-процессорные серверы с Xeon SPR их обгоняют, поскольку EPYC 9004 не поддерживают такие многопроцессорные конфигурации.

Тесты производительности HPCG. В отличие от рассмотренных выше данных о производительности в вычислительно-интенсивном HPL, HPCG с самого начала был ориентированным на суперкомпьютеры и связанным памятью тестом (см., например, [**313**]), и считался лучше соответствующим реальным приложениям. Устремления HPCG на эталонность реализовались в виде присутствия на сайте TOP500 отдельного списка суперкомпьютеров по производительности в HPCG, причем в последней доступной при написании обзора ноябрьской версии 2024 года[36] продолжает лидировать суперкомпьютер Fugaku с гомогенными узлами на базе ARM A64FX. Поскольку суперкомпьютеры без GPU не потеряли актуальность, рассмотрим производительность Xeon на тесте HPCG сразу за данными о производительности в HPL.

Данные о производительности HPCG (в GFLOPS) для разных серверов Dell PowerEdge с Xeon SPR (с 32-ядерными моделями Xeon 6430 и 8452Y и 56-ядерной моделью Xeon 8480+) представлены на рисунке 35.

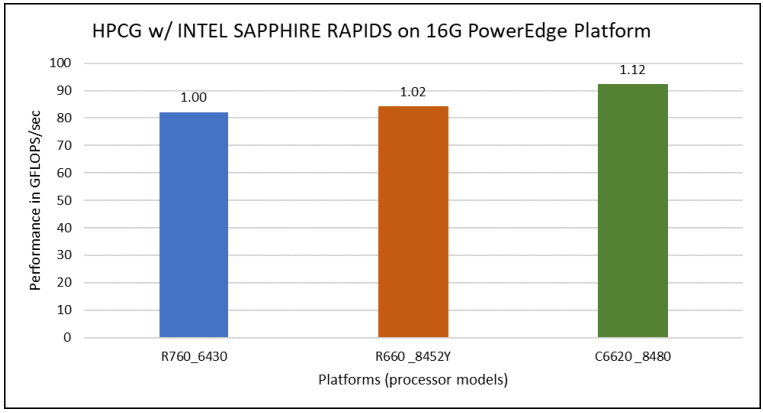


Рисунок 35. Производительность в HPCG в 2P-серверах с Xeon SPR (рисунок из [**314**])

Эти данные можно использовать с точки зрения относительной производительности разных моделей Xeon SPR, поскольку точный размер задачи для этого рисунка в [**314**] не приведен.

Однако имеются данные о производительности в HPCG для более широкого набора процессоров, включая Xeon EMR и EPYC 9004 [**202**, **289**], в том числе проведенные в рамках PTS-теста HPCG 3.1 [**202**], которые приведены в таблице 34.

[36] https://top500.org/lists/hpcg/list/2024/11/, accessed 7.05.2025.

Таблица 34. Производительность серверов с процессорами Xeon SPR, Xeon EMR и EPYC 9004 в тесте HPCG

Модель	Число ядер в процессоре	Производительность сервера, GFLOPS	
		1P-конфигурация	2P-конфигурация
Xeon 8592+	64	35.42	70.95
Xeon 8490H	60	31.24	60.42
Xeon 8380+		20.65	40.31
EPYC 9684X	96	23.78	44.11
EPYC 9654	96	32.13	43.97
EPYC 9684X	96		С детерминизмом Power: 73.26
EPYC 9654	96		С детерминизмом Power: 50.86

Примечание: время выполнения 60 сек., размерность подсетки 144.

Естественно, в том числе из-за более высокоскоростной памяти, Xeon SPR (модель 8490H) по данным этой таблицы сильно опережает Xeon ICL (модель 8380), а Xeon EMR (модель 8592+) быстрее Xeon SPR. При работе с параметрами BIOS в EPYC 9004 по умолчанию старшие модели Xeon SPR и Xeon EMR по производительности в HPCG существенно опередили EPYC 9004. При включении в BIOS для EPYC 9004 детерминизма Power, дающего достижение более высоких тактовых частот за счет повышения TDP (см. раздел 2.2 выше) производительность EPYC 9004 существенно возрастает. Однако и Xeon EMR, и Xeon SPR при этом по-прежнему существенно быстрее часто используемого в суперкомпьютерах процессора EPYC 9654, и лишь 2P-сервер с EPYC 9684X немного опередил 2P-сервер с Xeon 8592+.

Здесь встает вопрос уровня масштабирования производительности HPCG с числом ядер в многоядерных процессорах (для 96-ядерных EPYC 9004 в таблице 35 это более актуально, чем для Xeon) при использовании в [**202**] в качестве размерности локальной подсетки 144 для всех трех координат (образующаяся подсетка присваивается каждому MPI-процессу, см. [**315**]). В [**314**] для Xeon 8480+ была получена более высокая производительность, чем в [**202**] для Xeon 8592+, и там применялся размер подсетки не менее 192. По данным AMD [**121**] достигаемая производительность старших моделей EPYC 9004 в HPCG с размерностью подсетки 192 в пару раз выше, чем в этих PTS-тестах.

Параллельные тесты NAS (NPB). Графики зависимости производительности в этих тестах для 32-ядерных моделей Xeon SPR приведены выше на рисунках 15 и 16.

Здесь мы ограничимся анализом данных [202], где приведены данные о производительности и энергоэффективности в соответствующих PTS-тестах NPB для разных поколений масштабируемых процессоров Xeon, в том числе Xeon ICL, Xeon SPR и Xeon EMR, а также EPYC 9004 Genoa и Bergamo.

В таблице 35 приведены данные о производительности 1P- и 2P-серверов с Xeon 8490H, Xeon 8592+, а также EPYC 9654, EPYC 9684X и EPYC 9554. В ней приведены данные только для EPYC Genoa, поскольку в HPC обычно используются именно они, а не Bergamo с менее вычислительно эффективными ядрами Zen 4c. Но 128-ядерные EPYC 9754 с Zen 4c по производительности в NPB также, как и старшие модели Genoa, опережали Xeon SPR и Xeon EMR [202].

Таблица 35. Производительность сравниваемых поколений Xeon и EPYC в NPB 3.4

Модель	Число процессоров и ядер	Производительность, MOP/s	
		BT.C	SP.C
Xeon 8490H	1×60	180757	72703
	2×60	336632	152641
Xeon 8592+	1×64	236491	111687
	2×64	437264	242574
EPYC 9554	1× 64	233622	119472
	2×64	452577	242455
EPYC 9654	1×96	261865	125374
	2×95	509783	263543
		551536(*)	272752*
EPYC 9684X	1×96	312136	208538
	2×96	625625	353106
		708203(*)	390558*

(*) помечены результаты EPYC, проведенные с детерминизмом Power (остальные – в режиме по умолчанию).

В таблице были отобраны данные из [202], в которых не наблюдались грубые нарушения масштабирования производительности с числом ядер (этому может способствовать и связанность памятью тестов NPB). Согласно данным [202], во всех тестах EPYC Genoa опережали по производительности Xeon SPR и Xeon EMR, как в 1P- так и в 2P-конфигурации, причем иногда это было и при одинаковом числе ядер в Xeon и EPYC.

Это имело место как в детерминизме по умолчанию, так и в Power-режиме, который давал обычно существенное увеличение производительности Genoa. По производительности на Вт эти модели Xeon SPR и Xeon EMR также отставали от EPYC 9004 [202].

Имеется[37] более широкий набор данных о производительности разных моделей процессоров в PTS-тестах NPB. Некоторые данные для теста NAS FT.C (с трехмерным БПФ) приведены выше в таблице 17.

В этом тесте процессоры Xeon SPR и Xeon Max отстают в производительности от старших моделей EPYC 9004 (это имеет место и для Xeon EMR), а иногда и от EPYC Milan. Но в PTS-тестах NPB также нужно обращать внимание на соответствие используемых классов (размерностей задач) масштабированию производительности с числом ядер. В тесте LU.C, где отсутствие масштабирования производительности с ростом числа ядер внешне не проявлялось, 1P- и 2P-серверы с EPYC 9684X, EPYC 9654 и EPYC 9554 были быстрее аналогичных серверов с Xeon SPR и Xeon EMR, а 1P- и 2P-серверы с 32-ядерными EPYC 9374F опережали по производительности серверы с 60-ядерными EPYC 8490H (см. также обсуждение производительности EPYC 9004 в PTS-тестах NPB в разделе 2.4.5 выше).

Тесты производительности Graph500. Рассматривавшиеся немного выше тесты были ориентированы в первую очередь на задачи HPC, и актуальны в том числе и для суперкомпьютеров. Среди потенциально интересных и для современных суперкомпьютеров тестов можно отметить анализ производительности для интенсивной обработки данных, который проводится в тесте Graph500 для задачи работы с графами. Соответствующие данные для PTS-тестов Graph500 3.0 на 1P- и 2P-серверах с Xeon 8592+, Xeon 8490H и Xeon ICL 8380 представлены в [**202**]. Там видно, что Xeon SPR по производительности существенно опережает Xeon ICL, а Xeon EMR – соответственно опережает Xeon SPR. Но все процессоры Xeon при этом существенно отстают по производительности от старших моделей EPYC 9004.

Тесты производительности для облачных технологий. Разные тесты производительности, условно интегрированные в этом обзоре в группу актуальных для облачных технологий (как это было сделано ранее и при рассмотрении производительности EPYC 9004), естественно включают и онлайн аналитическую обработку (OLAP). Особенности для OLAP использующих чиплеты современных многоядерных процессоров (в первую очередь NUMA) рассмотрены в [**316**]. Там получены, в частности, данные о производительности TPC-H для 2P-сервера с 56-ядерными Xeon 8480+, а также 2P-сервера с 64-ядерными EPYC Milan 7713 и однопроцессорного сервера с 64-ядерным ARM-процессором AWS Graviton 3.

[37] https://openbenchmarking.org/test/pts/npb, accessed 12.03.2025.

Данные об ускорении в TPC-H, полученном на 2P-сервере с 64-ядерными Xeon 8592+ по сравнению с аналогичным сервером с 56-ядерными Xeon 8480+, представлены в [**305**]. Для теста OLTP-2, являющегося разработанным Fujitsu определенным аналогом стандарта TPC-E, в [**303**] приведены оценки результатов для серверов этой фирмы с разными моделями Xeon SPR (от 8- до 32-ядерных) и Xeon EMR (от 8- до 24-ядерных).

Другими актуальными тестами, отнесенными здесь к данной группе, являются стандартные тесты SAP[38]. В [**63**] были получены данные о производительности на 8-процессорном сервере Fujitsu с 60-ядерными Xeon 8490H для всех трех фаз теста SAP Business Warehouse (SAP BW) для базы данных SAP HANA. Для 2P-сервера Fujitsu с Xeon 8490H в [**166**] были получены данные о производительности в двухуровневом стандартном тесте SAP SD, и достигнуто ускорение по сравнению с 2P-сервером с 40-ядерными Xeon ICL 8380 в 1.6 раза (что близко к отношению числа ядер в этих серверах). В [**305**] приведены данные о производительности в SAP BW для HANA, полученные на 2P-сервере с 64-ядерными Xeon 8592+.

Тесты работы с базами данных также могут быть актуальны для облачной технологии. Здесь мы укажем на данные PTS-тестов PostgreSQL в [**202**] о производительности с Xeon 8592+ по сравнению с 56-ядерными Xeon 8490P+ и старшими моделями EPYC 9004. В этих тестах 2P-серверы с указанными процессорами дают число транзакций/с меньше, чем 1P-конфигурации. Что касается 1P-серверов, то по отношению к серверу с Xeon 8490H процессор Xeon 8592+ дает ускорение на 6.6%, а более быстрый 96-ядерный EPYC 9654 – на 10.8%.

Производительность задач механики сплошных сред. Относительная производительность 40-ядерного Xeon ICL 8380 и 56-ядерного Xeon SPR 8480+ в тестах программных CFD-средств OpenFOAM, CFD-приложений ANSYS Fluent, LS_DYNA и Mechanical, а также приложения прогноза погоды WRF представлена выше на рисунке 32, и для Xeon SPR показывает рост примерно в полтора раза, как возросло и число ядер.

Поскольку часто производительность CFD-приложений лимитируется умножением разреженной матрицы на вектор, в [**317**] на 2P-сервере с 56-ядерными Xeon 8480+ для задач CFD исследованы возможности преобразования таких умножений (SpMV) в произведения разреженных матриц (SpMM) для достижения более высокой производительности.

[38] https://www.sap.com/dmc/exp/2018-benchmark-directory/#/q2c, accessed 16.03.2025.

PTS-тест производительности LULESH (мини-приложения для решения задачи взрыва Седова), как отмечено выше в разделе 2.4.7, плохо масштабируется с числом ядер EPYC 9004 (при переходе от модели с меньшим числом ядер к более старшей модели с большим числом ядер, например, 2P-сервер с 64-ядерным EPYC 9554 работал быстрее, чем с 96-ядерным EPYC 9654 [**202**]). В этом тесте лучшую производительность показал 2P-сервер с 64-ядерными Xeon 8592+, который в 1.55 раза опередил 2P-сервер с EPYC 9654, и в 1.61 раза – с 60-ядерным Xeon 8490H. И по производительности на Вт серверы с Xeon 8592+ опередили серверы со старшими моделями EPYC 9004 [**202**].

В PTS-тесте OpenFOAM 10 drivaerFastback с маленьким размером сетки [**318**] 2P-сервер с Xeon 8592+ по производительности опередил все серверы с EPYC 9004, кроме 2P-сервера с EPYC 9684X, отстав от последнего всего на 4%. Но при среднем размере сетки более часто применяемый EPYC 9654 также опередил Xeon 8592+, на 10%. В таблице 36 представлены данные [**318**] о времени соответствующих расчетов для среднего размера сетки.

Таблица 36. Время расчета на 1P- и 2P-серверах с разными моделями процессоров в PTS-тесте OpenFOAM drivaerFastback со средним размером сетки

Модель	Число ядер	В конфигурации 1P	В конфигурации 2P
Xeon 8592+	64	302.1	137.0
Xeon 8490H	60	420.6	196.5
Xeon Max 9480	56	410.3	186.3
EPYC 9654	96	325.4	124.3
EPYC 9684X	96	178.2	93.6

Процессор Xeon 8592+ существенно превосходит здесь по производительности Xeon 8490H, что во многом связано, вероятно, с более быстрой используемой памятью. Производительность возрастает также за счет применения HBM-памяти (в Xeon Max 9480) или за счет применения 3D V-cache (в EPYC 9684X), но 96-ядерные модели EPYC 9004 опережают Xeon безотносительно к наличию таких усовершенствований в иерархии памяти. Данные таблицы демонстрируют ожидаемое хорошее масштабирование производительности при переходе от 1P- к 2P-конфигурации.

В отчете [**319**] для PTS-теста мотоцикла в OpenFOAM был продемонстрирован эффект применения режима OPM в BIOS для Xeon 8592+. Время расчета в этом режиме было примерно на 1 секунду больше, но потребление мощности Xeon в среднем для 2P-конфигурации было почти

на 80 Вт ниже, а пиковое потребление мощности – на 117 Вт ниже. Для более вычислительно сложной модели drivaerFastback производительность в режиме OPM была сопоставима с исходной, но с OPM среднее потребление мощности 2P было на 33 Вт ниже, а пиковое потребление мощности – на 37 Вт меньше. Но оба этих теста проводились с маленьким размером сетки.

В PTS-тесте WRF 4.2.2 с «conus 2.5km» в качестве исходных данных в [**318**] 2P-сервер с Xeon 8592+ также, как с рассмотренным выше тестом OpenFOAM drivaerFastback с маленьким размером сетки опередил по производительности 2P-серверы с Xeon 8490H (на 19%) и с EPYC 9654 (на 24%), то есть последний отстал и от 2P-сервера с Xeon 8490H. Но лидером по производительности здесь был 2P-сервер с EPYC 9684X, обогнавший сервер с Xeon 8592+ на 26%. Кроме того, 1P-серверы с EPYC 9684X и EPYC 9654 были быстрее 1P-сервера с Xeon 8592+ (хотя EPYC 9654 – только на 6%). EPYC 9654 опередил при этом Xeon 8490H на 18%.

В [**289**] в аналогичном тесте WRF с «conus 2.5km» (там имеются также данные для 2P-сервера с 56-ядерными Xeon 8480+) 2P-сервер с Xeon 8592+ опередил 2P-сервер с EPYC 9654, но отстал по производительности от 2P-сервера с имеющим столько же ядер (64) EPYC 9554. Все эти тесты требуют прояснения ситуации в первую очередь с рассмотрением зависимости производительности от числа задействуемых в расчете ядер.

На основании представленных выше данных PTS-тестов LULESH, OpenFOAM и WRF видно существенное увеличение производительности Xeon 8592+ по сравнению с Xeon 8490H. Можно сказать о конкурентоспособности по производительности Xeon 8592+ и старших моделей EPYC 9004. Xeon 8592+ иногда опережал по производительности EPYC 9654, но это относилось к задачам скорее небольших размерностей. Имеющиеся данные важно уточнять с точки зрения определения зависимости производительности от числа задействуемых в серверах ядер.

Масштабирование производительности OpenFOAM и WRF от числа содержащих Xeon 8480+ узлов кластера с Infiniband NDR 200 рассмотрено в [**320**].

Производительность в задачах вычислительной химии. Далее мы рассмотрим информацию о производительности Xeon SPR и Xeon EMR в задачах молекулярной динамики, квантовой химии, а также условно отнесенного здесь к вычислительной химии мини-приложения miniBUDE для молекулярного докинга. Наибольшее количество таких данных традиционно относится к молекулярной динамике, в том числе из-за часто высоких показателей масштабирования производительности с ростом числа используемых ядер.

Для приложения молекулярной динамики GROMACS, которое часто считают наиболее хорошо распараллеленным, данные о производительности в PTS-тесте GROMACS 2023 для 1P- и 2P-конфигураций серверов с 64-ядерными Xeon 8592+, 60-ядерными Xeon 8490H и 40-ядерными Xeon 8380+ получены в [**202**] для исходных данных water_GMX50_bare (расчет комплексов из большого числа молекул воды) с применением MPI-распараллеливания. Соответствующая производительность (временной интервал движения в нс, рассчитанный за день) представлена в таблице 37, где она сопоставлена и со старшими моделями EPYC 9004.

Таблица 37. Сопоставление производительности GROMACS (в нс/день) в серверах с Xeon SPR, Xeon EMR с серверами на других процессорах

Модель	Число ядер	В конфигурации 1P	В конфигурации 2P
Xeon 8592+	64	10.5	18.3
Xeon 8490H	60	8.6	15.2
Xeon 8380	40	4.8	9.0
EPYC 9654X	96	11.5	18.8
EPYC 9654	96	11.4	19.0
EPYC 9554	64	9.7	16.9

Данные этой таблицы говорят о высоком росте производительности при переходе от старших моделей Xeon ICL к старшим моделям Xeon SPR, а затем к Xeon EMR. При переходе к Xeon SPR достигнутый рост производительности превысил увеличение числа ядер. Аналогичное имеет место при переходе от Xeon SPR к Xeon EMR.

Как для представленных в таблице моделей Xeon, так и для моделей EPYC 9004 имеет место хорошее масштабирование производительности при переходе от 1P- к 2P-конфигурации серверов. Серверы со старшими моделями EPYC 9004 опередили по производительности серверы со старшими моделями Xeon благодаря большему числу ядер. Производительность в этом тесте возрастала и при дальнейшем росте числа ядер в модели. Еще более высокая, чем представленная в таблице производительность, была получена в [**202**] на сервере со 128-ядерными EPYC 9754. Кроме того, производительность серверов с EPYC 9004 еще несколько повышается при включении детерминизма Power.

При сопоставлении серверов с одинаковым числом ядер с 64-ядерными процессорами Xeon 8592+ и с EPYC 9554 производительность Xeon в этом тесте процентов на десять выше. Однако данных для расчета с EPYC 9554 с детерминизмом Power в [**202**] не представлено.

По данным [**202**], с точки зрения энергоэффективности процессоры Xeon уступали старшим моделям EPYC 9004.

Другие данные о производительности для серверов с Xeon 8592+ и Xeon 8490H, представленные в таблице 37, были получены в [**318**] для PTS-теста GROMACS 2024 (с GROMACS 1.9) для тех же исходных данных с тем же распараллеливанием MPI, и также сопоставлены с EPYC 9004. Эти результаты близки к приведенным в таблице 37, и здесь не рассматриваются.

Имеется[39] более широкий охват данных о производительности в этих же вариантах PTS-тестов GROMACS для разных процессоров Xeon, так и вообще для других процессоров. Эти данные показывают, что 1P-серверы с Xeon 8592+ и Xeon 8490H опережают по производительности 1P-серверы с современными процессорами ARM – со 192-ядерным AmpereOne и с 96-ядерным Neoverse-V2 (вероятно, это AWS Graviton 4).

Что касается данных о производительности Xeon SPR и Xeon EMR в другом широко распространенном приложении молекулярной динамики, NAMD, то производительность 2P-сервера с Xeon 8490H в тесте гомодимера протеинов Her1-Her1, содержащего 456 тысяч атомов, представлена в [**321**].

По данным AMD [**195**], 2P-серверы со 128-ядерными процессорами Bergamo (EPYC 9754) сильно опережали по производительности 2P-серверы с 56-ядерными Xeon 8480+ в нескольких тестах с разными исходными данными, что может быть естественно при хорошей масштабируемости с числом ядер. Про существенно более высокую производительность NAMD в 2P-сервере с EPYC 9654 по сравнению с сервером на Xeon 8480+ AMD сообщила в [**196**] без указания на конкретные исходные данные теста. Данные о производительности серверов с некоторыми моделями Xeon SPR в PTS-тестах NAMD имеются[40].

Имеются[41] данные о производительности Xeon SPR и Xeon EMR в PTS-тестах LAMMPS версии 1.4, другого известного приложения молекулярной динамики, с расчетами протеина родопсин и молекулярной системы из 20 тысяч атомов. Хотя согласно этим результатам Xeon EMR и Xeon SPR отставали по производительности от старших моделей EPYC 9004, это требует дополнительного более тонкого анализа достигаемой производительности, в том числе с исследованием зависимости от числа используемых в расчетах ядер.

---

[39] https://openbenchmarking.org/test/pts/gromacs, accessed 23.03.2025.

[40] https://openbenchmarking.org/test/pts/namd, accessed 24.03.2025.

[41] https://openbenchmarking.org/test/pts/lammps, accessed 24.03.2025.

Для PTS-теста NWChem 7.0.2, относящегося к характеризующемуся высокоразвитыми средствами распараллеливания квантовохимическому программному комплексу NWChem, в [**202**] получены данные о производительности 1P- и 2P- серверов с Xeon EMR и Xeon SPR, в том числе с Xeon 8592+ и Xeon 8490H. Эти данные относятся к DFT-расчету молекулы C240, который обсуждался выше в разделе 2.4.8 с точки зрения производительности EPYC 9004. Серверы с Xeon 8592+ здесь опережают по производительности серверы с Xeon 8490H примерно на десять процентов. Серверы с EPYC 9654 здесь также существенно опередили по производительности системы с Xeon 8592+. Но как было указано в разделе 2.4.8, этот тест с рассматриваемыми в обзоре процессорами отличается плохой масштабируемостью с числом задействованных в тесте ядер, и, в частности, при переходе от 1P- к 2P-конфигурациям. Здесь, например, 1P-сервер с 64-ядерным EPYC 9554 показал более высокую производительность, чем системы с 96-ядерным EPYC 9654.

Доступен[42] более широкий набор данных о производительности разных процессоров в этом тесте. По этим данным, 1P-сервер с 72-ядерным ARMv8 Neoverse-V2 (вероятно, Nvidia Grace) потребовал меньше времени расчета в этом тесте, чем 64-ядерный EPYC 9554.

Другим квантовохимическим приложением, для которого имеются данные о производительности с 56-ядерным Xeon 8480+, является CP2K. В [**320**] для 16-узлового кластера с Infiniband NDR 200 продемонстрирована хорошая масштабируемость производительности CP2K в методах DFT и RI-MP2. Для приложения Quantum ESPRESSO в [**195**] AMD сообщила, что 2P-сервер с Xeon 8480+ отставал по производительности от 2P-сервера со 128-ядерными EPYC 9754 примерно в 1.4 раза. Строго говоря, этот расчет относится к квантовой молекулярной динамике методом Кар-Парринелло, но время вычисления здесь лимитируется расчетом сил квантовохимическим методом DFT.

В тестах производительности современных вычислительных систем часто используется мини-приложение молекулярного докинга, miniBUDE. В PTS-тесте miniBUDE с OpenMP-распараллеливанием данные о производительности получены в [**202**], в частности, для 1P- и 2P-систем с Xeon 8592+ и Xeon 8490H. Производительность этого теста хорошо масштабируется при переходе на более старшие модели с большим числом ядер и от 1P- к 2P-конфигурации. Далее указываемая производительность

[42] https://openbenchmarking.org/test/pts/nwchem, accessed 13.12.2024.

относится к 2P-серверам. 64-ядерный Xeon 8592+ оказался быстрее, чем 60-ядерный Xeon 8490H в 1.38 раза, что гораздо больше, чем увеличение числа ядер. Однако Xeon 8592+ отстал по производительности от EPYC 9654 с детерминизмом Power в 1.88 раза, в то время как число ядер у Xeon в полтора раза меньше, чем у EPYC 9654.

По энергоэффективности в тесте miniBUDE вычислительные системы с EPYC 9004 в [**202**] также сильно опередили серверы с Xeon EMR и Xeon SPR.

Далее будет рассматриваться производительность Xeon SPR и Xeon EMR в задачах ИИ. Поэтому представляется разумным дать здесь небольшое заключение по данным о производительности в широко используемых тестах, включая тесты приложений. Можно сказать, что Xeon EMR (большее число данных относилось к Xeon 8592+) существенно опережает по производительности старшие модели Xeon SPR и имеет также стоимостные преимущества.

При сопоставлении производительности Xeon EMR и EPYC 9004 нужно, конечно, ориентироваться на конкретную область применения. Но в целом очень грубо можно сказать, что по производительности (в первую очередь в HPC-области) старшие модели Xeon EMR и соответственно Xeon SPR обычно отстают от имеющих большее число ядер старших моделей EPYC 9004. При сопоставлении серверов с близким или одинаковым числом ядер Xeon EMR нередко достигали более высокой производительности, но при этом отставали от EPYC 9004 с точки зрения энергоэффективности и стоимости.

4.2. Производительность Xeon SPR и Xeon EMR в задачах ИИ

Данные о производительности для задач ИИ вычислительных систем (от серверов до небольшого кластера) на базе Xeon SPR в сопоставлении с аналогичными результатами для 3-го поколения масштабируемых процессоров Xeon и для процессоров Zen 3 представлены Intel в [**322**].

Информация об ускорении Xeon SPR за счет применения в рабочих нагрузках ИИ возможностей AMX имеется в [**118**]. В предыдущем разделе приведены данные о производительности Xeon SPR в актуальных для ИИ операциях GEMM в формате BF16, поддерживаемом AMX. В конце этого раздела производительность GEMM также проиллюстрирована для форматов FP32 и BF16 в сопоставлении с процессорами ARM.

Среди публикаций, нацеленных на повышение производительности в ИИ с использованием AMX, укажем работы [310, 323–326]. Понятно, что достигаемая величина ускорения бывает разная. Например, в [325] доработки в программном обеспечении с использованием AMX давали ускорение до 1.5 раз. В [326] с применением для больших языковых моделей генерации с расширенным поиском (Retrieval-Augmented Generation) с использованием точного поиска ближайших соседей, которое является связанным памятью, ускорение за счет AMX составляло от 2 до 10%. С использованием GPU Nvidia H100 ускорение было от пяти (с одним H100) до сорока раз (с 8 H100), хотя в [326] отмечена высокая стоимость аппаратных средств с H100 (для варианта с емкостью памяти в 512 ГБ потребовалось 8 GPU). Но преимущества Xeon SPR благодаря использованию AMX очевидны, а Xeon EMR дает дальнейшее существенное улучшение.

Intel полагает, что задачи вывода моделей ИИ будут использоваться небольшими компаниями для легких моделей ИИ (а не для массивных моделей более чем с триллионом параметров), для которых может быть достаточно всего нескольких ядер современных процессоров Xeon [327]. Этому отвечает и разработка Intel программных средств OpenVINO [328], ориентированных на высокую производительность в первую очередь вывода.

Другой иллюстрацией того, где для ИИ возможно эффективно применять серверы с Xeon без GPU, могут быть ориентированные на задачи ИИ тесты MLPerf-Inference [215] (имеются в виду некоторые современные результаты этих тестов). Эти тесты стали стандартом и очень активно используются в научных исследованиях последних лет в области ИИ. Соответствующие результаты представлены в [329].

В серверных сценариях этих тестов (для онлайн-приложений со случайным поступлением запросов, где при создании вывода учитывается общая целевая величина задержки [329]), имеются представленные Intel результаты, где 2P-сервер с 64-ядерными Xeon 8592+ без GPU опережал результаты с современными GPU Nvidia. Так, в тесте модели рекомендаций dlrm-v2-99.9 (последнее число – уровень точности в процентах) сервер Intel дал 949.2 запроса/сек, а данные Nvidia с GPU H200-SXM-141GB-CTS указывали 155.0 запросов/с. Число запросов/сек здесь указывает число обработанных при соблюдении требования к границе задержки запросов.

В тесте большой языковой модели (gpt-j-99.9 в серверном сценарии) с Xeon 8592+ Intel был получен наивысший на 14.01.2025 результат,

949.20 токена/с, а в одном из результатов от Supermicro с 8 GPU Nvidia H100-SXM-80GB – 900.0 токенов/с. В конце лета 2024 года Nvidia объявила о получении кардинально более высоких показателей для таких тестов вывода моделей ИИ [**330**], но соответствующие результаты на сайте тестов MLperf на момент написания обзора пока не были представлены.

Пропускная способность является метрикой для автономных сценариев в тестах MLPerf-Inference [**215**] (для приложений пакетной обработки, где все данные доступны немедленно, а задержка не ограничена). Данные о производительности серверов с Xeon 8592+ для автономных и серверных сценариев Llama-2b-99 и Llama-2b-99.9 на условиях тестов MLPerf-inference имеются в [**329**]. Там, например, в наборе тестов версии 4.1 есть полученный Intel для серверного варианта Llama-2b-99 на сервере с Xeon 8592+ результат, который на 8% выше полученного Dell показателя для сервера с 8 GPU AMD MI300X (данные на 14.01.2025).

Данные Dell о производительности ее серверов с Xeon EMR в MLPerf-Inference 4.0 имеются в [**331**], а с Xeon SPR в больших языковых моделях с применением форматов FP32, FP16 и BF16 – в [**332**].

Ясно, что для приведенных выше задач вывода процессоры Xeon EMR являются актуальными. Другой иллюстрацией этого можно считать и работу [**333**], в которой предложен усовершенствованный подход к ускорению вывода больших языковых моделей с применением распараллеливания, в результате чего при приемлемых величинах достигнутой пропускной способности и задержки с применением Xeon EMR 6538N энергопотребление было снижено на 49% по сравнению с применением сервера с GPU Nvidia A100.

Хотя выше приведены отдельные примеры задач ИИ, где показаны успехи производительности Xeon по сравнению с системами с GPU, это, конечно, всего лишь иллюстрация того, что такие задачи бывают. Безусловно, на огромном количестве других тестов для ИИ, обычно требующих больше вычислительных ресурсов, процессоры x86, в том числе и Xeon, сильно отстают по производительности от современных GPU.

После того, как AMD предоставила на выставке Computex 2024 видео с указанием опережения по пропускной способности вывода большой языковой модели новейшими 128-ядерными процессорами Turin (архитектуры Zen 5, см. раздел 7 далее) 64-ядерных процессоров Xeon 8592+ без указания использовавшегося программного стека и соглашений об уровне обслуживания (SLA), Intel продемонстрировала противоположные

результаты [334]. Обе информации появились еще до начала поставок AMD этих процессоров Zen 5 Turin. В 2P-сервере с Xeon 8592+ при использовании Llama2-7B с форматом INT4, SLA 50 мс при процентиле P99 (т.е. когда только 1% запросов обрабатываются за время больше задержки) с применением расширения Intel PyTorch (IPEX) с открытым исходным кодом Intel была получена пропускная способность 686 токенов/с против 671 для 2P-сервера со 128-ядерными процессорами Turin.

Intel показала в [334] также гораздо более высокую пропускную способность вывода с форматом INT8 для нескольких моделей глубокого обучения в 2P-серверах с Xeon 8592+ и с Xeon 8480+ по сравнению с 2P-серверами с содержащими больше процессорных ядер EPYC 9654 и 9754.

В [335] получены интересные данные об уменьшении (с ростом числа используемых ядер) времени вывода для моделей ResNet-50, Yolo [336] и средств OpenVINO при использовании IPEX на 2P-сервере с 56-ядерными Xeon 8480+ (см. рисунок 36).

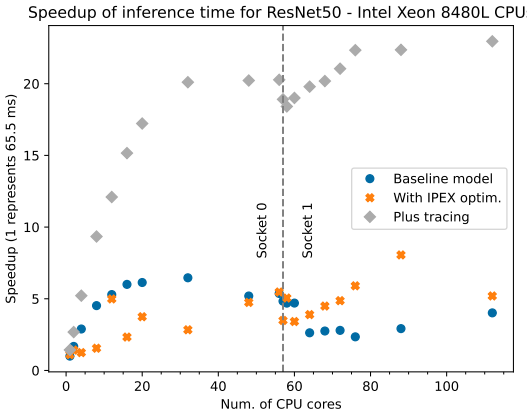


Рисунок 36. Время вывода для модели ResNet-50. Синим цветом приведены базовые показатели, желтым — полученные при использовании IPEX (рисунок из [335])

В таблице 38 приведены данные PTS-тестов для серверов при использовании ориентированных на повышение производительности вывода программных средств Intel OpenVINO.

Здесь не приведены результаты тестов задержек, поскольку они часто демонстрируют более высокие результаты вообще на более простых (не на серверных) процессорах x86. Автор не располагает данными о применении AMX на процессорах Xeon в этих тестах.

Таблица 38. Производительность (результатов/с) в PTS-тестах вывода OpenVINO 2024.0

	N	Тест 1	Тест 2	Тест 3	Тест 4	Тест 5	Тест 6
Xeon 8592+	2P	5338		1130	**546**	**32006**	
	1P	2808	2597	592	**284**	**16716**	406
Xeon 8490H	2P	4093	4396	796	425	28502	653
	1P	1902	1541	477	168	8944	299
Xeon Max 9480	2P	2940	3567	812	380		535
	1P	1804	1541	476	218		301
EPYC 9684X	2P	**8279**	5196	1112	206	10610	**1107**
	1P	**4139**	2553	583	99	4972	**475**
EPYC 9654	2P	7812	5032	1015	197	10237	767
	1P	3185	**2609**	546	96	5007	352
EPYC 9754	2P	6474	**5824**	**1146**	241	12294	759
	1P		2504	**621**	122		277

Данные из [https://openbenchmarking.org/test/pts/openvino 17.01.2025].
Приведены средние значения разных выполнений тестов.
Жирным цветом обозначены максимальные для приведенных процессоров показатели.

N – число процессоров в сервере.

Тест 1 – обнаружение транспортных средств. Формат: FP16

Тест 2 – распознавание рукописного английского текста. Формат: FP16

Тест 3 – перевод с английского на немецкий. Формат: FP16

Тест 4 – распознавание лиц. Формат: FP16-INT8

Тест 5 – обнаружение пористости сварного шва. Формат: FP16

Тест 6 – обнаружение человека. Формат: FP16

Хотя в некоторых таких тестах старшие модели Xeon EMR и Xeon SPR опережали EPYC 9004, последние чаще оказывались впереди (см. также таблицу 22). Другие данные с PTS-тестом OpenVINO 2023.2.dev для таких моделей Xeon приведены в [**202**]. Там тоже в одних случаях быстрее были серверы с Xeon 8592+, в других – со старшими моделями EPYC 9004.

Следует также отметить, что нужно достаточно аккуратно оценивать смысл появляющихся многочисленных данных о производительности x86 в самых разных тестах для вывода моделей ИИ. В качестве примера укажем на данные производительности вывода (кадров в секунду) при использовании обычных для фреймворка TensorFlow тестов (для старших моделей рассматриваемых в обзоре процессоров)[43].

[43] https://openbenchmarking.org/test/pts/tensorflow, accessed 4.12.2024.

Аналогичные данные о производительности имеются, например, в [**200**] и [**289**]. Для таких тестов с использованием модели ResNet-50 при работе с размерами пакетов (BS) 64 и 256 (там есть также данные для Xeon 6 и EPYC Zen 5) возникают ситуации, когда при переходе от более младшей модели процессора с меньшим числом ядер к более старшей модели с большим числом ядер, при переходе от 1P- к 2P-конфигурации сервера производительность падает или плохо масштабируется.

В [**200**] приведены данные таких тестов при BS=512, но и в аналогичных данных[44] имеются, может более слабые, эффекты плохой масштабируемости с числом ядер. Грубо говоря, в таких тестах более высокую производительность можно достигать в серверах с более дешевыми и менее производительными процессорами с меньшим числом ядер.

Сопоставление производительности таких процессоров в этих тестах носит весьма ограниченный смысл. Сравнение рассматриваемых процессоров величинами задержки в таких тестах менее интересно. Ясно, что в таких тестах следовало бы проводить исследование зависимости производительности от числа используемых ядер.

Большое количество результатов тестов производительности и энергоэффективности 2P-сервера с Xeon 8592+ для выводов и обучения, с применением самых разных моделей ИИ, разных используемых фреймворков и форматов чисел, для разных размеров пакетов представлено Intel в [**337**], а для 2P-сервера с 56-ядерными Xeon 8480+ – в [**338**]. Для обучения там использовался один процессор.

Там же можно найти также аналогичные данные для 32-ядерных Xeon SPR 6448Y, и для масштабируемых процессоров Xeon третьего поколения, а также для Intel GPU Flex Series 140 и 170. Эти данные позволяют получить, в частности, оценки прогресса в производительности для задач ИИ при переходе на более новые поколения масштабируемых процессоров Xeon.

Такие данные об ускорении производительности в 2P-сервере с Xeon 8592+ по сравнению с Xeon 8480+ фирма Intel представила для задач вывода и обучения при применении фреймворка TensorFlow для разных моделей ИИ с форматами BF16 и INT8 и с разными BS [**339**]. BS=1 на рисунке 37 для вывода отвечает использованию реального времени, а BS=x означает применение оптимального по производительности размера пакета. Ускорение вывода при работе с Xeon 8592+ по сравнению с Xeon 8480+ составляет от 23 до 39%.

[44] https://openbenchmarking.org/test/pts/tensorflow, accessed 4.12.2024.

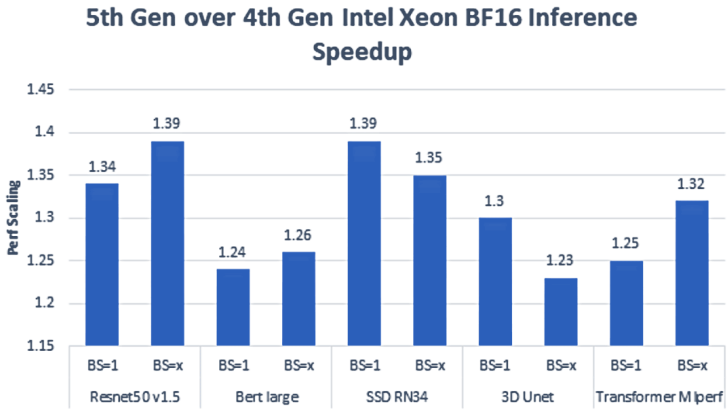


Рисунок 37. Ускорение на Xeon 8592+ по сравнению с Xeon 8480+ в выводе моделей ИИ (рисунок из [**339**])

На рисунке 38 показано улучшение производительности с Xeon 8592+ относительно Xeon 8480+ для случая обучения моделей со смешанной точностью BF16/FP32. Видно, что ускорение при этом составляет от 6 до 26%.

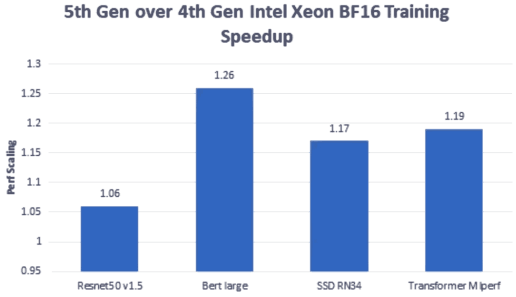


Рисунок 38. Ускорение Xeon 8592+ по сравнению с Xeon 8480+ для обучения ИИ (рисунок из [**339**])

Выше уже приводился пример эффективности Xeon EMR для ИИ с точки зрения энергопотребления. В [**340**] для 2P-сервера с 64-ядерными Xeon 8592+ проведено исследование энергоэффективности при использовании разных моделей ИИ, в том числе больших языковых моделей, модели распознавания изображений ResNet-50 и модели обработки естественного языка Bert-Large.

Для этого применялись возможности усовершенствованной в Xeon EMR функции оптимизированного режима питания OPM, в которой используются тонкие инструменты, в том числе изменение частоты внутреннего межсоединения. За счет работы OPM было достигнуто увеличение энергоэффективности до 25%,но улучшение возможно только в случае низкой нагрузки на процессор (не более 50%).

В заключение рассмотрения производительности Xeon SPR и Xeon EMR в задачах ИИ укажем данные из работы [**311**], которая исходно ориентирована на создание общих для задач HPC и ИИ переносимых программных ядер для разных аппаратных платформ с использованием примитивов тензорной обработки (Tensor Processing Primitives, TPP). В ней представлены, в частности, данные о производительности 2P-серверов с 56-ядерными Xeon 8480+ и 64-ядерными ARM AWS Graviton 3 (с ядрами Neoverse V1) для задач умножения матриц, интересных в первую очередь для глубокого обучения (GEMM в форматах BF16 и FP32).

Соответствующие данные представлены на рисунке 39; они для разных программных вариантов GEMM (PARLOOPER относится к реализации в [**311**]) демонстрируют, в частности, большое преимущество в производительности процессоров Xeon SPR над ARM при сопоставимом числе ядер.

Процессоры Xeon SPR и особенно Xeon EMR, безусловно, представляют интерес для применения их возможностей для ИИ. Они имеют определенные преимущества по производительности по сравнению с EPYC 9004 благодаря аппаратным особенностям (в первую очередь из-за поддержки AMX) и эффективным средствам программного обеспечения от Intel для задач ИИ (целый ряд их уже упоминался ранее).

Можно найти много данных, демонстрирующих преимущество производительности для задач ИИ серверов с Xeon 8490H или с Xeon 8592+ над серверами со старшими моделями EPYC 9004, например, в PTS-тесте с OneDNN 3.4 с конфигурацией Harness: Deconvolution Batch shapes_3d[45], с OneDNN 3.3 в [**202**]. В данных таких тестов OneDNN также надо контролировать выбранные конфигурации, поскольку они могут давать неподходящее для таких процессоров масштабирование с числом ядер.

[45] https://openbenchmarking.org/test/pts/onednn, accessed 6.02.2025.

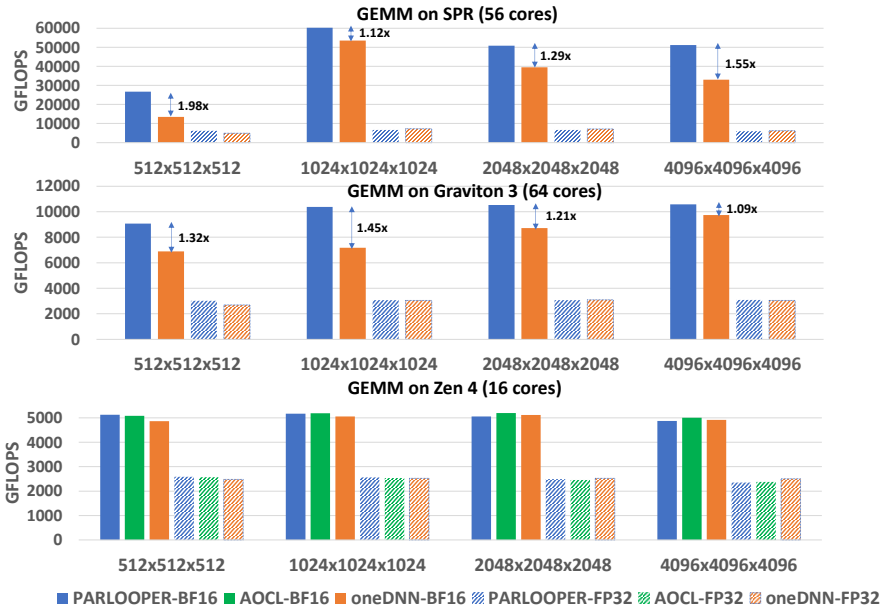


Рисунок 39. Производительность разных реализаций GEMM (рисунок из [311]); данные «GEMM on Zen 4» не относятся к рассматриваемым здесь серверным процессорам

Здесь и ранее приводились и примеры, когда старшие модели EPYC 9004 в задачах ИИ опережали по производительности соответствующие модели Xeon SPR или EMR. Можно указать также на данные AMD [341] о более высокой производительности 2P-сервера с EPYC 9654 по сравнению с сервером с Xeon EMR 8592+ при использовании средств XGBoost (eXtreme Gradient Boosting) – масштабируемого высокопроизводительного метода для машинного обучения при его применению с наборами данных по прогнозированию задержки авиарейсов или по обнаружению бозонов Хиггса.

Однако во многих этих результатах не имеется данных о применении при этом оптимизированных именно для новых процессоров Xeon программных средств Intel. В любом случае ясно, что для задач ИИ конкурентоспособность Xeon SPR и особенно Xeon EMR по сравнению с EPYC 9004 больше, чем в других широко используемых областях применения (которые обсуждались выше в разделе 4.1).

4.3. Сопоставление отдельных видов производительности Xeon SPR

В этом разделе приведены сравнительные данные о производительности Xeon SPR и EPYC 9004 на моделях среднего класса, а также о сравнении производительности Xeon SPR и GPGPU.

Производительность моделей условного среднего класса Xeon SPR и EPYC 9004. В HPC-центре университета Юта были проведены расчеты в разных тестах на серверах с разными моделями EPYC 9004 и Xeon SPR [**100**], и показано, что среди моделей среднего класса производительность Xeon SPR (32-ядерный Xeon 6430) обычно несколько уступает аналогичному EPYC 9004 (32-ядерному EPYC 9334), но имеет при этом заметно более низкую цену. Кроме сопоставления серверов со средними моделями, проведены тесты производительности и вариантов с 1P-конфигурациями с применением старших моделей EPYC 9004, в том числе с тем же числом (64) ядер, что в 2P-моделях.

Нацеленность [**100**] на модели среднего класса была обусловлена предположением, что они могут быть более актуальны для многочисленных HPC-расчетов, в том числе в небольших кластерах. А, кроме того, имеющиеся массовые данные результатов PTS-тестов на сайте openbenhcmarking.org в основном относятся к старшим моделям этих процессоров. Можно предположить, что к средним моделям процессоров в [**100**] отнесены имеющие цену около $3000 и ниже.

В [**100**] проведены интересные в этом плане тестирования производительности, включая HPL, HPCC, NPB и приложений вычислительной химии (LAMMPS, GROMACS и NWChem). Определенным минусом этих данных является применение ориентированного в первую очередь на HPC и суперкомпьютеры пакетного менеджера Spack [**342**] с параметрами построения исполняемых модулей, не ориентированных узко на соответствующие модели процессоров.

Некоторые данные из [**100**] приведены в других разделах обзора, здесь мы ограничимся в основном результатами тестов HPCC, приведенными в таблице 39.

Из [**100**] видно, что эти модели Xeon и EPYC с одинаковым числом ядер показывают конкурентоспособные данные по производительности. В некоторых тестах Xeon 6430 по производительности опережали EPYC 9334, в т.ч. в HPL и DGEMM. Правда, при использовании библиотеки MKL (для EPYC 9334 – с включенным режимом, дающим максимальную производительность), в HPL полученная производительность с EPYC (3.53 GFLOPS) была выше, чем с Xeon 6430 (3.44 GFLOPS).

Таблица 39. Данные о производительности в тестах HPCC

Модель		Xeon 6430	EPYC 9334
Число ядер		2×32	2×32
Частота[1], ГГц		2.1	2.7
Цена		$2138	$2990
TDP,Вт		270	210
HPL, TFLOPS		3.23	3.11
DGEMM Single GFLOPS		71.14	59.87
PTRANS ГБ/с		18.20	29.85
Random Access, GUPs	Single	0.107	0.148
	MPI	0.337	0.445
FFTE MPI, GFLOPS		57.73	89.58
Stream triad, Single ГБ/с		13.28	43.20

Данные из [100]. Single относится к тестам на одном ядре.
[1] Базовая величина.

Можно обратить также внимание и на данные тестов SPEC CPU2017 fp_speed и fp_rate [51]. По данным на 13.02.2025 максимальные достигнутые результаты для 2P-серверов с EPYC 9334 составляли 339/358 и 843/845 для fp_speed и fp_rate соответственно (приведены базовый/пиковый показатели). Аналогичные результаты для Xeon 6430 составляют 295/295 и 683/703 соответственно. Для 1P-серверов в этих тестах EPYC 9334 также оказались быстрее.

Xeon 6430 имеют существенно более низкую стоимость, хотя при этом имеют более высокий TDP, чем EPYC 9334. Из указанных выше данных можно сделать предположение о более высокой конкурентоспособности (в отношении процессоров EPYC 9004) модели Xeon 6430 по сравнению со старшими моделями Xeon SPR.

В таблице 40 мы привели данные тестов SPECcpu 2017 с плавающей запятой для процессоров Xeon ICL/SPR/EMR и EPYC Genoa среднего класса. 64-ядерный EPYC 9554P приведен здесь в качестве иллюстрации возможной альтернативы двум 32-ядерным EPYC Genoa. Все приведенные в таблице процессоры (кроме этого 64-ядерного) можно отнести к среднему классу с числом ядер около 30 и стоимостью меньше $3000. Соответственно здесь представлена самая младшая 32-ядерная модель EPYC Genoa (32-ядерная модель EPYC 9354 стоит уже больше $3000).

Таблица 40. Данные тестов SPECcpu 2017 для EPYC 9004, Xeon ICL, Xeon SPR и Xeon EMR

Процессор	EPYC 9334		EPYC 9554P	Xeon 6330		Xeon 6430		Xeon 6530	
Число ЦП	1	2	1	1	2	1	2	1	2
Число ядер	32	64	64	28	56	32	64	32	64
Цена	$2990	$5980	$7104	$2128	$4256	$2138	$4276	$2128	$4256
SPECcpu 2017 fp_speed peak/base	253/239[1]	358/339[2]	308/301[3]	116/114[4]	219/219[5]	186/186[6]	295/295[7]	–/220[14]	326/326[15]
SPECcpu 2017 fp_rate peak/base	422/420[8]	845/843[9]	665/618[8]	188/179[10]	423/406[11]	330/318[12]	703/683[13]	377/369[6]	782/765[14]
Частота, ГГц	2.7–3.9		3.1–3.75	2–3.1		2.1–3.4		2.1–4	
TDP	210 Вт	420 Вт	360 Вт	205 Вт	410 Вт	270 Вт	540 Вт	270 Вт	540 Вт

Данные на 3.01.2025. Цены из [49], для Xeon 6330 – минимальная из [https://www.intel.com/content/www/us/en/ark.html#@PanelLabel595 23.02.2025]. Отправители и использованные в расчете системы:

[1] ASUS RS520A-E12-RS12U;
[2] ASUS RS720A-E12-RS12;
[3] Lenovo ThinkSystem SR635 V3;
[4] HPE ProLiant DL110 Gen10;
[5] xFusion FusionServer 1288H V6;
[6] HPE ProLiant DL320 Gen11;
[7] ASUS ESC4000-E11;
[8] xFusion FusionServer 1158H V7;
[9] xFusion FusionServer 2258 V7;
[10] HPE ProLiant DL110 Gen10 Plus;
[11] Tyrone Camarero TDI100C3R-212;
[12] Dell Precision 7960 Rack;
[13] ZTE R5300G5;
[14] Lenovo ThinkSystem SD530 V3;
[15] xFusion FusionServer 1288H V7;
[15] ASUS RS720-E11-RS12U.

Согласно данным этой таблицы, среди процессоров среднего класса EPYC обгоняют по производительности Xeon при одинаковом числе ядер. Хотя по отношению стоимость/производительность Xeon могут быть лучше, но это требует дополниптельного анализа. Кроме того, надо учитывать и величины TDP.

Сравнение производительности Xeon SPR с GPGPU. В заключение в этом разделе полезно привести также данные об относительной производительности Xeon SPR на приложениях HPC по сравнению с современными GPGPU, чтобы оценить возможную эффективность их применения. Для этого используем информацию для сервера с двумя Xeon 8480+, к которому добавлялись GPU Nvidia A100, H100 и H200 [**343**] (данные на 9.10.2024). Мы отобрали данные для двух известных HPC-приложений в области CFD (FUN3D) и молекулярной динамики (GROMACS). В [**343**] представлены аналогичные данные и для других программ молекулярной динамики (AMBER, NAMD и LAMMPS), с расчетами разных молекулярных систем. Выбор GROMACS здесь произведен из-за частого восприятия этого приложения как наиболее высокопроизводительного и достигающего более хорошее распараллеливание.

При добавлении в сервер одного GPU A100 производительность FUN3D в тесте dpw_wbt0_crs-3.6Mn_5 возросла в 4 раза, а в GROMACS в тесте STMV для молекулярной системы, содержащей около миллиона атомов – в 2 раза. При добавлении вместо A100 самого современного GPU H200 производительность этих приложений возрастает еще раза в два. Интересно также, что при использовании вместо этого сервера варианта с Nvidia GH200 производительность практически не меняется [**343**], то есть использование интегрированного процессора Grace в этом плане не отличается от применения пары Xeon 8480+.

По данным [**344**], в приложении NAMD в тесте Her1-Her1 производительность 2P-сервера с 60-ядерными Xeon 8490H уступала производительности одного GPU Nvidia Geforce RTX 6000 Ada более чем в четыре раза.

Сравнение энергоэффективности [**345**] сервера с Xeon 8480+ относительно применения там GPU Nvidia H100 или с интегрированным GH200 на приложении молекулярной динамики NAMD или программных средствах MILC для теории сильных взаимодействий субатомной физики показывает, безусловно преимущество применения GPU. Однако все это может служить только численной иллюстрацией для оценок целесообразности их применения при наличии такой возможности.

Приведем еще три сравнивающих Xeon SPR и GPU примера. В [**346**] сопоставлена производительность в CFD-приложении OpenFOAM 2P-сервера с 56-ядерными Xeon 8480+ и GPU Intel Max 1550. Сравнение

производительности другого 2P-сервера с Xeon 8480+ для задач CFD с применением OpenFOAM и средств ИИ при работе с разными числовыми форматами и с применением смешанной точности по сравнению с одним GPU Intel Max 1550 проведено в [347]. В [335] изучено время вывода в 2P-сервере с Xeon 8480+ или с GPU Intel Max 1100 в моделях ИИ, в том числе в ResNet-50 с анализом зависимости от числа использованных ядер в Xeon.

В заключение этого подраздела следует отметить, что получение выигрыша в производительности за счет применения GPU может требовать большой работы и соответственно не всегда реализуется. Так, в [283] в приложении магнитогидродинамики MHD 2P-сервер с Xeon SPR с HBM-памятью (Xeon Max 9480) давал производительность, близкую к GPU Nvidia A100. Производительность Xeon Max и рассматривается далее.

4.4. Производительность процессоров Xeon Max

Хорошей общей демонстрацией сравнительной производительности Xeon Max с Xeon ICL и Xeon SPR является приведенный выше рисунок 32, в котором сравниваются данные для 2P-серверов с 56-ядерными Xeon Max 9480, Xeon 8480+ и 40-ядерными Xeon 8380.

Ранее в обзоре в процессе рассмотрения данных о производительности процессоров других архитектур многократно приводились данные и о производительности Xeon Max, в том числе в относительном виде (сопоставительного характера).

Появление процессоров x86 с поддержкой HBM-памяти (Xeon Max – вторые в мире после A64FX процессоры, поддерживающие HBM и первые, поддерживающие одновременно HBM и DDR) во время современного расширения лимитирования производительности вычислительных систем пропускной способностью памяти породило разные надежды и вызвало естественный всплеск числа публикаций о производительности Xeon Max в разных областях науки (см., например, [3, 163, 278, 283, 284, 348–351]), включая работы и самих сотрудников Intel [352]. Отдельного внимания заслуживает публикация, в которой рассматривается применение Xeon Max в узлах суперкомпьютера Aurora, и приводится информация о производительности приложений из области HPC [353].

Некоторые из возникших ожиданий не оправдались, хотя они могли быть и потенциально явно завышенными. Так, в [348] указали, что памяти HBM оказалось недостаточно для полного преодоления разрыва в производительности моделирования термоядерного синтеза между центральными и графическими процессорами.

Возможные причины ограничения в ожидавшемся росте производительности рассмотрены далее. Главными вопросами здесь являются достигаемые показатели памяти (в первую очередь пропускная способность), поскольку HBM достаточно дорогая (цены на разные модели Xeon Max см. выше в таблице 36).

Понятно, что после выхода на рынок Xeon Max сразу стали появляться работы с исследованиями достигаемых задержек и пропускной способности памяти HBM, см., например, [**284**]. Анализ зависимости задержки и пропускной способности HBM в 1P-конфигурации с Xeon Max 9470C от различных конфигураций памяти проведен в [**353**]. Мы далее сосредоточимся на наиболее важном показателе, пропускной способности, которую обычно измеряют в тестах stream.

Пропускная способность памяти Xeon Max в тесте stream. Влияние разных режимов работы с HBM-памятью на пропускную способность в 2P-сервере с 56-ядерными Xeon Max 9480 относительно пропускной способности в 2P-сервере с 60-ядерными Xeon 8490H, соединенными только с памятью DDR5, было показано в [**280**] и представлено на рисунке 40. Скорее всего, там использован вариант stream triad, хотя это в [**280**] явно

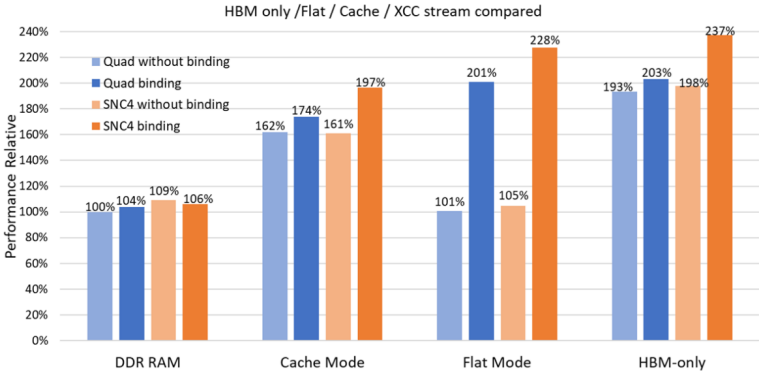


Рисунок 40. Относительная пропускная способность памяти Xeon Max 9480 в разных режимах (рисунок из [**280**])

не указано. На этом рисунке иллюстрируется также влияние привязывания к доменам NUMA, что для Xeon Max помогает увеличить достигаемую пропускную способность (quad здесь означает работу с квадрантом, когда вся HBM-память процессора Xeon Max образует один NUMA-узел).

Режим кэширования ожидаемо показал самую низкую пропускную способность, а в режиме сочетания HBM-only с SNC4 пропускная способность самая большая. В сочетании плоского режима с кластеризацией квадрантом в [**280**] исследована зависимость от числа задействованных ядер, от

двух до 56; соответствующие данные представлены на рисунке 41. Они показывают, что хорошее масштабирование пропускной способности наблюдается только до 28 ядер, достигая при этом 96% от максимальной получавшейся величины.

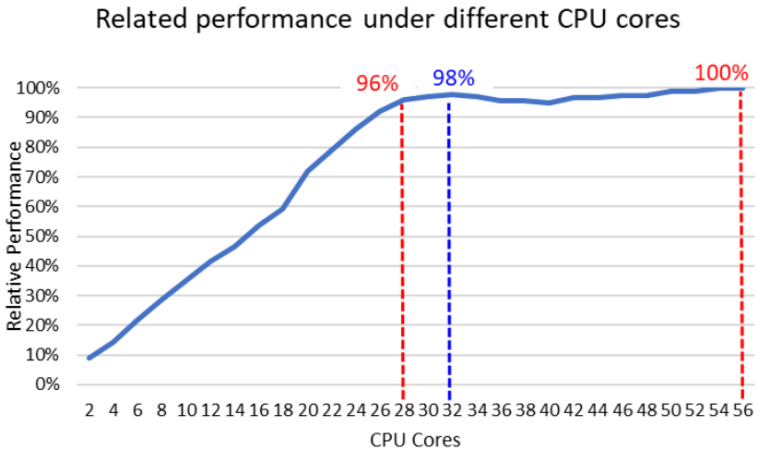


Рисунок 41. Зависимость достигаемой пропускной способности памяти от числа используемых ядер (рисунок из [**280**])

Логарифмическое масштабирование числа ядер в аналогичных результатах на рисунке 33 (в разделе 4.1), вероятно, не дает возможности увидеть подобную [**280**] зависимость пропускной способности.

При этом надо отметить, что по данным [**283**] максимальная пропускная способность при работе с DDR5 достигалась практически уже при 7 ядрах.

Пропускная способность в тесте stream triad для 2P-сервера с 48-ядерными Xeon Max 9468 в актуальном, позволяющем достичь более высокие показатели, плоском режиме с SNC4 была измерена в [**14**]. Полученное там значение, около 1361 ГБ/с, оказалось в 1.6 раз больше, чем для 1P-сервера с ARM A64FX.

Важное подробное исследование особенностей работы с HBM-памятью в Xeon Max, включая пропускную способность в тестах stream, проведено в работах автора самих тестов stream МакКалпина [**281**, **354**].

Иллюстрацией этого является рисунок 42, который показывает, что в тесте stream даже идеальное для пропускной способности распараллеливание на ядрах Xeon Max позволило бы достичь только 68% от пиковых 1638 ГБ/с для HBM. Эти данные относятся к 1P-серверу с 56-ядерным Xeon Max 9480. Соответственно исследовалась проблема пропускной

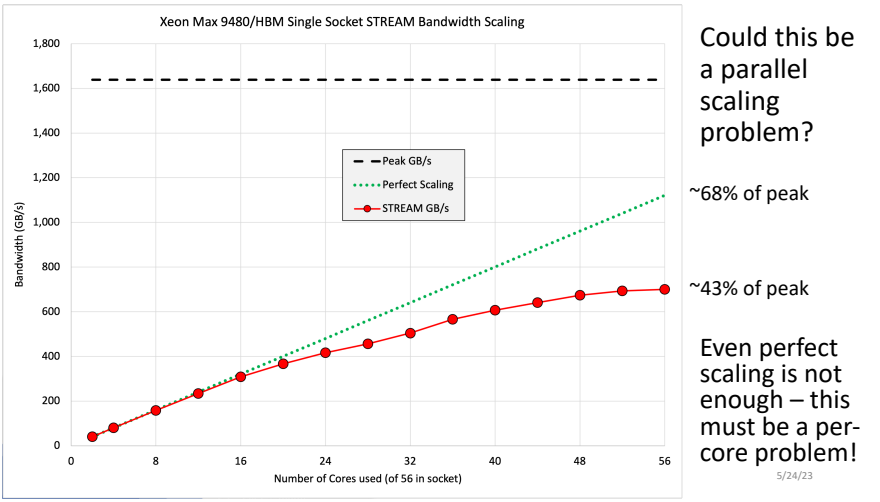


Рисунок 42. Пропускная способность HBM в Xeon Max 9480 в зависимости от числа задействованных ядер (рисунок из [354])

памяти на уровне ядра. Само распараллеливание, как показывают данные этого рисунка, сильно далеко от идеала.

Изучение возможного распараллеливания при обработке промахов в кэшах L1 и L2 с учетом задержки в соответствии с законом Литтла из теории очередей дало предельную пропускную способность одного ядра около 24 ГБ/с, а измеренные величины показали 20 ГБ/с в stream и 22 ГБ/с в тесте «только чтения» из памяти. Но реально измеренная пропускная способность процессора (на всех ядрах) была гораздо меньше прогнозируемой в соответствии с этой теорией.

Тонкий анализ показателей Xeon Max во время выполнения тестов показал, что двумерное решеточное межсоединение в этих процессорах обеспечивает только половину пропускной способности, необходимой для достижения полной возможной пропускной способности чтения из HBM [281, 354].

Несмотря на достижение менее половины возможной пропускной способности HBM из-за имеющихся ограничений параллелизма при обработке промахов в кэш-память ядрами Xeon Max и пропускной способности их межсоединения, процессоры Xeon Max обеспечивают более высокую пропускную способность HBM по сравнению с DDR от 2.5 до 3.5 раз (в тестах «только чтения» и stream triad соответственно) [281]. Так что важным является рост реальной производительности, достигаемый при увеличении цены за счет применения HBM.

После публикаций [**281**, **354**] в некоторых других работах также отмечались определенные недостатки применения HBM в Xeon Max. Так, в [**283**] указано на низкую эффективную пропускную способность на ядро.

Для 1P- и 2P-серверов с Xeon Max 9480 в тесте BabelStream triad зависимость достигаемой пропускной способности от размера используемых в тестировании массивов (от 19.2 МБ до 9.8 ГБ) в режиме HBM-only была исследована в [**163**]. Максимальная пропускная способность достигалась при размерах не более нескольких сот мегабайт, а затем она уменьшалась. Говорить о пропускной способности именно памяти можно лишь при достаточно больших размерах массивов, когда измеренное значение стабилизировалось; при меньших размерах получаемая величина относится к работе кэша [**163**].

Соответственная стабилизированная величина для 2P-сервера составила 1446 ГБ/с при использовании оптимальных для других исследованных тестов ключей компиляции, и 1643 ГБ/с при применении специального оптимизированного для stream набора ключей (streaming store) из Intel OneAPI Base & HPC toolkits 2023.1. Эти величины составляют соответственно 55% и 63% [**163**] от пиковой пропускной способности, в качестве которой там использовано значение 1.3 ТБ/с на сокет со стороны возможностей процессора, с учетом задержек кэш-памяти [**354**]. Полученные процентные показатели соответствует приведенным выше выводам [**281**, **354**].

AMD в [**160**] продемонстрировала среднеарифметические величины пропускной способности от всех четырех компонент stream для 2P-серверов с 32-ядерными Xeon Max 9462 и EPYC 9384X с 3D V-cache для разного числа используемых нитей от 1 до 64 в виде относительных показателей. Во всех использованных количествах нитей (степени двух) пропускная способность в EPYC была выше, чем у Xeon Max в режимах кэширования и HBM-only, за исключением последнего при 64 нитях. Однако в [**160**] не указаны размеры использовавшихся массивов, что может означать возможное достижение преимущества в пропускной способности не собственно памяти, а с 3D V-cache.

Производительность Xeon Max в тестах SPEC. В работе [**355**] изучено влияние HBM на производительность Xeon Max в тестах SPECcpu 2017 и SPEChpc 2021. Далее здесь рассматриваются только официальные утвержденные результаты тестов SPEC.

Данные о производительности процессоров Xeon Max в тестах SPECcpu 2017, fp_speed и fp_rate, приведены выше в таблице 37. Мы сопоставим здесь производительность Xeon Max с Xeon SPR и Xeon EMR при одинаковом числе ядер. При большем числе ядер Xeon SPR или EMR безусловно опережают Xeon Max в этих тестах.

Для 2P-серверов с одинаковым числом ядер (по 56 на процессор) переход от Xeon 8480+ на работу с HBM в Xeon Max 9480 повышает производительность в пиковом показателе SPECfp_rate на 4%. Возможно и из-за немного более низких показателей частот ядер у Xeon Max 9480, в SPECfp_speed он уступает Xeon 8480+. При этом 56-ядерный Xeon 8570 без наличия HBM-памяти имеет более высокую производительность в обоих тестах SPECfp 2017 при более низкой стоимости, так что прогресс в технологии Intel и технологии DDR-памяти оказался в этом плане для Xeon EMR существенно важнее перехода на дорогую HBM-память. В 5-м поколении масштабируемых процессоров Xeon, как и в Xeon 6, HBM-память Intel не предлагается.

Xeon Max в этих тестах отстают по производительности от старших моделей EPYC 9004. Результаты тестов SPECcpu 2017 для целочисленной арифметики не представляют интереса для целей данного обзора. Соответствующие данные не меняют картины относительной производительности Xeon Max и EPYC 9004.

Интереснее производительность Xeon Max в SPEChpc 2021. Для серверов и кластеров с Xeon Max 9480 для всех вариантов тестов от tiny до large соответствующие данные представлены на сайте[46]. В этих данных на 1.04.2025 2P-серверы с Xeon Max 9480 уступают по производительности 2P-серверам с EPYC 9654.

Представляющие актуальность для данного обзора данные о производительности Xeon Max, которые обсуждаются далее, относятся всего к двум моделям, 48-ядерной Xeon Max 9468 и 56-ядерной Xeon Max 9480 (данные об этих моделях приведены в таблице 36 выше). В публикациях, в которых рассмотрена производительность в широком классе тестов или приложений, обычно обсуждается только по одной из этих моделей. Поэтому анализ производительности для удобства изложения далее также разбивается на две части по разным моделям. Каждый из соответствующих разделов включает различные тесты производительности. Во втором из этих разделов производится также необходимое обобщение.

4.4.1. *Производительность 48-ядерной модели Xeon Max 9468*

Поскольку Xeon Max имеет HBM-память, которая ранее стала применяться в A64FX, появились и сопоставляющие вычислительные системы на их базе публикации. Так, в [**356**] для глубокого обучения (многослойного персептрона) исследована производительность и энергопотребление в Infiniband-кластерах с 2P-узлами на базе Xeon Max 9468 и 1P-узлами с A64FX, и изучено их масштабирование в зависимости от числа ядер и узлов кластера.

Вообще о производительности Xeon Max 9468 имеются интересные данные, которые включают и ее сопоставления с различными процессорами ARM (см. таблицу 41).

[46] https://www.spec.org/cgi-bin/osgresults?conf=hpc2021, accessed 1.04.2025.

Таблица 41. Данные о производительности Xeon Max 9468 в сопоставлении с ARM-процессорами

Тест производительности (единицы измеряемой производительности)[1]	Исходные данные	Intel Xeon Max 9468, 96 ядер[2]		Fujitsu A64FX, 48 ядер	Amazon Graviton 3, 64 ядра	Nvidia Grace, 144 ядра[3]
		DDR	HBM			
DGEMM (GFLOPS)		4787	5392	1978	1/158	4461
HPL (GFLOPS)		2211	2862	1177	965	3124
FFT (GFLOPS)		129	143	24.4	71.0	134.2
HPCG (GFLOPS)		84	198	64.4		106.5
OpenFOAM (время выполнения, минут)	MotoBikeQ, 11M cells	18.39	14.87			13.87
	MotoBikeX6, 35M cells	53.45	39.65			
GROMACS (нс/день)	MEM 82K атомов	203.6	206.1	22.8	71.4	171
	RIB 2M атомов	13.9	13.5			12.7
	PEP 12M атомов	1.2	1.2			0.977

В таблице использованы данные из [14, 357].

[1] Первые 3 приведенных в таблице теста взяты из HPCC.

[2] Объяснение преимущественно HBM-варианта и DDR приводится в тексте далее.

[3] Приводятся максимальные представленные в [357] величины, полученные с использованием разных программных средств.

Что касается использованного для тестов Xeon Max 9468 преимущественного HBM-варианта, то его установили с помощью указания ключа для numactl, -preferred-many, для всех NUMA-доменов памяти HBM. Как указано в [14], это давало в их тестах такую же производительность, как при использовании привязки к доменам NUMA.

Хотя в некоторых случаях в этой таблице данные относятся к 1P-серверам (с A64FX, Graviton 3), Xeon Max 9468 показал преимущества в производительности по сравнению со всеми современными ARM-процессорами (хотя в некоторых случаях для этого требовалось активное применение HBM). То же самое имеет место в производительности на ядро (если посчитать ее, исходя из данных этой таблицы).

Исключением оказался тест OpenFOAM, где производительность имеющего в полтора раза больше ядер процессора Nvidia Grace (использующего 32 канала LPDDR5X) оказалось на 7% выше, чем у Xeon Max 9468 с HBM. По производительности на ядро Xeon Max здесь впереди. Надо также учитывать, что при хорошем масштабировании производительности в тесте Xeon Max 9480 должен показывать более высокие результаты, и что в Xeon SPR есть более высокопроизводительные модели. В Xeon EMR достигается дополнительное повышение производительности.

Для задач ИИ (в тесте AI Benchmark Alpha, в таблице он не представлен по причине отсутствия данных для Xeon Max) даже более старый Xeon ICL 6330 как правило опережал ARM-процессоры по производительности [357].

Из данных таблицы 41 видно, как более высокая пропускная способность HBM-памяти сказывается на росте производительности на вычислительно-интенсивных тестах (например, DGEMM), и как этот эффект усиливается на тестах, связанных памятью (например, HPCG).

Если сопоставить производительность 48-ядерных Xeon Max 9468 в DGEMM и HPL с производительностью EPYC 9004 (см. таблицу 15), то видно, что эта модель Xeon Max опережает 32-ядерные EPYC 9004, и отстает от 64-ядерных EPYC 9554, а тем более от содержащих большее число ядер процессоров EPYC. В требующем большой пропускной способности HPCG с эффективно использующим HBM вариантом (197.5 GFLOPS) сильно опережает представленные AMD данные в [121] (максимальный показатель - 137.9 GFLOPS для EPYC 9684X).

Однако для сравнения необходимо сопоставлять данные для расчета с одинаковыми исходными данными (в случае с HPCG – например, размеры сеток), что для вышеуказанных чисел для HPCG неизвестно. Соответственно далее мы рассматриваем только сопоставимые данные о производительности в тестах PTS с сайта openbenchmarking.org. Здесь полезно напомнить, что (как указано выше в разделе 2.4.4) 64-ядерный Xeon SPR 8462Y в PTS-тестах HPCG 3.1 опережал EPYC 9004.

Ряд данных, демонстрирующих производительность в разных PTS-тестах Xeon Max 9468 в сравнении с Xeon Max 9480, приводится в следующем разделе 4.4.2.

Если посмотреть на данные PTS-тестов - NPB, задач механики сплошных сред, где приложения часто связаны памятью (CloverLeaf, OpenFOAM, WRF), задач молекулярной динамики (NAMD, LAMMPS, GROMACS) и молекулярного докинга (miniBUDE) на сайте openbenchmarking.org, то по данным на декабрь 2024 года Xeon Max 9468 практически всегда уступает по производительности старшим моделям EPYC 9004.

Но в PTS-тестах для задач ИИ ситуация другая. В вариантах тестов OpenVINO, которые можно отнести к обработке изображений (данные на 17.01.2025), Xeon Max 9468 часто опережал 96-ядерные EPYC 9654 и 9684X, хотя иногда отставал от 128-ядерного EPYC 9754.

В PTS-тесте TensorFlow с моделью ResNet-50 по данным на 18.01.2025 Xeon Max 9468 опережал EPYC 9004 при размере пакетов 256, но отставал при работе с более крупными пакетами (512). В некоторых PTS-тестах oneDNN (данные на 19.01.2025) Xeon Max 9458 также опережал 96-ядерные EPYC 9004, отставая от 128-ядерного EPYC 9754.

Однако эти проведенные PTS-тесты, упоминаемые в трех абзацах выше, часто отличаются плохим масштабированием производительности с числом используемых ядер. Здесь имеются в виду факты уменьшения производительности при ее сопоставлении для процессоров с меньшим числом ядер по сравнению с более старшими моделями с большим числом ядер, или при переходе от 1P- к 2P-конфигурациям. Это уменьшает значение соответствующих тестов.

Поскольку в следующем разделе еще проводится обсуждение производительности Xeon Max 9480 в сопоставлении с Xeon Max 9468, мы завершаем здесь анализ данной модели.

4.4.2. *Производительность 56-ядерной модели Xeon Max 9480*

Понятно, что для топ-модели данного семейства, Xeon Max 9480, появилось много данных о производительности. Так, в [**283**] показано, что 2P-сервер с Xeon Max 9480 в тестах DGEMM и HPL дает только 87% от производительности 2P-сервера с также 56-ядерными Xeon 8480+. Другие данные со сравнением производительности в HPL у 2P-сервера с Xeon Max 9480 и аналогичных серверов с Xeon SPR и Xeon EMR имеются в [**303**]. Там с Xeon Max 9480 получено 6781 GFLOPS, а с Xeon 8480+ - выше, 7293 GFLOPS (понятно, что 60-ядерная топ-модель Xeon SPR 8490H достигла еще более высокую производительность, 7584 GFLOPS). Соответствующие данные о производительности Xeon Max 9480 в сравнении с другими моделями Xeon SPR представлены выше в таблице 41.

Такое отставание от производительности Xeon 8480+, очевидно, связано с тем, что его ядра имеют более высокую тактовую частоту (как

базовую, так и турбо), чем Xeon Max 9480, и является иллюстрацией того, что применение Xeon Max может быть актуально только для связанных памятью приложений.

В PTS-тесте HPCG 3.1 с размерами подсеток 144 на 2P-сервере с Xeon Max 9480 в [**358**] была получена производительность 74.3 GFLOPS, что, однако, ниже, чем в полученных для Xeon Max 9468 данных в таблице 41.

В тесте БПФ (S-FFTE), где важна пропускная способность памяти, сервер с Xeon Max 9480 был быстрее, чем сервер с Xeon 8480+, в 1.46 раза. И в связанных памятью задачах магнитогидродинамики производительность Xeon Max 9480 была гораздо выше [**283**]. Вообще в мини-приложении магнитогидродинамики CloverLeaf 2D/CloverLeaf 3D производительность Xeon Max 9480 сильно выше, чем у Xeon 8480+ [**3**]. Электромагнетизм нужен для термоядерного синтеза, и HBM-память в Xeon Max 9480 в приложении CGYRO также дает значительный прирост производительности [**348**].

Еще одной демонстрацией преимуществ Xeon Max 9480 для связанных памятью приложений, работающих со структурированными или неструктурированными сетками, являются данные работы [**3**]. Кроме только что упоминавшихся мини-приложений CloverLeaf 3D и CloverLeaf 2D, там представлены данные о производительности и шести других приложений. Но все они предполагают применение специфических программных средств, OPS для структурированных сеток и OP2 – для неструктурированных.

В [**3**] получены данные не только по производительности, но и по энергопотреблению для различных 2P-серверов из Intel Developer Cloud (см. об этом в разделе 5), в том числе с Xeon Max 9480 (в режиме HBM-only + SNC4), Xeon 8480+ (с тем же числом 56 ядер на процессор), Xeon 8592+ и Xeon 6960P (Granite Rapids). На таких отобранных приложениях Xeon Max 9480 часто опережал по производительности не только Xeon EMR, но и Xeon 6.

Большое количество данных о производительности 2P-сервера с Xeon Max 9480 для самых разных приложений HPC, включая предсказание погоды (WRF), молекулярную динамику (NAMD, AMBER), квантовую химию (PARSEC – для конечно-разностных расчетов методом DFT) и другие, получено в [**355**]. Там проводится сопоставление производительности при отключении работы памяти HBM или DDR5.

Широкий набор данных о производительности и энергопотреблении 2P-серверов с Xeon Max 9480 и Xeon Max 9468 представлен в [**359**]. Эти данные интересны также тем, что там исследована производительность в трех разных режимах работы с HBM – HBM-only, режиме кэширования и режиме с деактивированной HBM-памятью. Но вместо типично применяемого для больших расчетов плоского режима с использованием дополнительной памяти DDR, что может требовать более тонкой работы, изучена производительность в варианте, когда применение HBM было заблокировано.

Некоторые из полученных в [**359**] данных представлены в таблицах 42 и 43.

Таблица 42. Производительность в HPC у 2P-серверов с Xeon Max

Тест	Производительность Xeon Max 9480			Производительность Xeon Max 9468		
	Без HBM	Кэширование с HBM	HBM-only	Без HBM	Кэширование с HBM	HBM-only
OpenFOAM, тест 1[1]	4613.9	3258.6	2908.8	4629.1	3236.3	2918.4
OpenFOAM, тест 2[2]	190.7	177.0	165.1	166.5	157.8	141.9
OpenFOAM, тест 3[3]	237.4	199.9	176.7	245.3	195.6	177.8
LULESH, z/s	44118.3	55339.5	60681.6	45058.0	56829.2	62348.3
NPB, SP.C	140910.0	196046.2	212262.2	132417.4	184462.3	196927.9
NPB, BT.C	279271.5	328222.2	351853.6	265661.2	307050.7	330958.0
NPB, LU.C	223823.1	273779.7	278606.5	234956.3	276681.9	284194.2
NPB, MG.C	164986.7	249731.4	253047.6	162971.3	249546.4	265747.5
Quantum ESPRESSO[4]	346.0	339.1	323.7	334.5	326.0	309.1
GROMACS[5]	11.4	12.1	13.2	11.0	12.3	12.9
miniBUDE[6]	2670.9	2799.7	3032.9	3026.7	3110.1	3234.8

Данные из [359]. Для тестов NPB производительность приведена в Mop/s.

[1] время выполнения (секунд) в тесте drivaerFastback, Large Mesh Size;

[2] время на решетку (секунд) в тесте drivaerFastback, Medium Mesh Size;

[3] время выполнения (секунд) в тесте drivaerFastback, Medium Mesh Size;

[4] время выполнения (секунд) в тесте AUSURF112

[5] количество рассчитанных нс/день в тесте water_GMX50_bare

[6] GFInst/s в тесте BM2

То, что производительность возрастает при переходе от применения DDR-памяти к кэшированию HBM-памятью и далее с работой только с HBM, очевидно. Понятно, что эффект должен быть больше для связанных памятью тестов. Однако данные этой таблицы позволяют оценить это количественно. Эффект перехода от 48-ядерной модели к 56-ядерной Xeon Max 9480 при нормальном масштабировании производительности также очевиден, но данные таблицы демонстрируют это количественно. При анализах надо иметь в виду, что тактовая частота Xeon Max 9468 выше, чем у Xeon Max 9480 на 10.5% (см. таблицу 46), а число ядер меньше на 17%.

Из четырех проведенных в [**359**] тестов NPB со средним размером проблемы (C) Xeon Max 9480 в режиме HBM-only, дающем наивысшую производительность, показал преимущество в ней по сравнению с более младшей моделью только в двух – SP.C и BT.C.

В соответствии с результатами таблицы 42, переход от 48-ядерной модели Xeon Max 9468 к 56-ядерной Xeon Max 9480 в тестах OpenFOAM, ориентированных на задачи CFD, наблюдается низкий прирост производительности. В тесте мини-приложения CFD, LULESH, при таком переходе производительность уменьшается.

В использовавшемся в [**359**] тесте Quantum Espresso[47], который согласно применялся в качестве одного из тестов на суперкомпьютере Fugaku, проводился расчет квантовохимическим методом DFT с применением псевдопотенциалов в базисе плоских волн для системы из 112 атомов золота. Однако данные таблицы показывают, что при переходе от Xeon Max 9468 к более старшей модели Xeon 9480 время расчета в этом тесте не уменьшается, а возрастает.

В известном высоким достигаемым уровнем распараллеливания программном комплексе молекулярной динамики GROMACS (недавно появилась работа [**360**] о его распараллеливании в кластере, содержащем в узлах 2P-серверы с 64-ядерными EPYC 9554, всего 65k ядер) прирост производительности при переходе на Xeon Max 9480 оказался всего два %. Для данных таблицы 42 использовался широко распространенный тест GROMACS, water_GMX50_bare, включающий расчеты комплексов воды, содержащих от 0.65 тысячи до трех миллионов атомов.

В мини-приложении молекулярного докинга (miniBUDE) данные [**359**] также показывают понижение производительности при переходе от Xeon Max 9468 к топ-модели Xeon 8480.

[47] https://www.hpci-office.jp/documents/appli_software/Fugaku_QE_performance.pdf, accessed 10.05.2025.

Для задач ИИ, несомненно, представляют интерес данные тестов производительности разработанных Intel средств OpenVINO. Этих тестов было много проведено в [**359**]. Мы исключили из анализа результаты тестирований на уровне задержек, поскольку они обычно отличаются плохим масштабированием, а более высокие показатели часто можно получать на десктопных процессорах. Отобранные из [**359**] данные приведены в таблице 43.

Таблица 43. Производительность (кадров/с) Xeon Max в PTS-тестах OpenVINO 2022.3

Тест	Xeon Max 9480			Xeon Max 9468		
Person Detection, FP16	33.3	35.2	37.3	33.8	34.5	36.6
Person Detection, FP32	33.7	35.7	37.9	33.1	34.2	36.0
Face Detection, FP16-INT8	293.5	329.6	359.6	285.3	328.0	354.2
Weld Porosity Detection, FP16	16321.9	16480.5	17508.3	15419.6	15767.4	16960.1
Weld Porosity Detection, FP16-INT8	31507.7	30761.6	33984.7	30066.7	31679.3	33063.8
Person Vehicle Bike Detection FP16	5792.1	6157.7	6613.6	5615.0	6224.4	6539.6

Выигрыш в производительности при замене Xeon Max 9468 на Xeon Max 9480 не превышает нескольких процентов. Сопоставительные данные о производительности Xeon Max 9480, Xeon SPR, Xeon EMR и EPYC 9004 приведены выше в таблицах 43 и 38. По этим данным, Xeon Max 9480 отставал в производительности от Xeon EMR и EPYC 9004, хотя часто опережал Xeon SPR 8490H.

Сами по себе приведенные выше отдельные данные о производительности Xeon Max 9480 по сравнению с более младшей моделью не говорят о каких-то недостатках в Xeon Max. Проблема в первую очередь в том, что в широко применяющихся тестах часто отсутствует анализ достигаемой производительности в зависимости от числа задействуемых в расчетах ядер, и не рассматривается ее зависимость от размера исследуемого объекта. Хотя нельзя отвергнуть и возможные проблемы с уровнем распараллеливания в Xeon Max, про которые говорилось в [**281**, **354**] в отношении тестов stream. В практическом плане можно ориентироваться на оценку [**355**], что увеличение стоимости за счет применения HBM по сравнению с Xeon SPR компенсируется, если производительность связанного памятью приложения возрастает более чем на 30%.

Отдельного сопоставления производительности Xeon Max с EPYC 9004 здесь не проводится, в том числе из-за прекращения Intel дальнейшей

поддержки HBM в Xeon EMR и Xeon 6. Аналогичные рассмотренному здесь данные о производительности этих процессоров AMD, в том числе и со сравнением производительности с Xeon Max, приведены выше в разделе 2.4. Многочисленные сравнительные данные о производительности EPYC 9004 и Xeon Max для большого количества различных PTS-тестов можно получить на сайте openbenchmarkung.org, а также из использующих эти тесты обзоров производительности (см., например, [**318**]). Старшие модели EPYC 9004 обычно опережали Xeon SPR и Xeon EMR по производительности, а последние в свою очередь чаще опережают Xeon Max, за исключением отдельных областей применения со связанными памятью программами.

5. О применении x86 для виртуализации и облачных технологий в области HPC и ИИ

Базирующиеся на виртуализации облачные технологии очень активно применяются и для задач HPC - в первую очередь, вероятно, из-за экономической эффективности. Облачные технологии применимы на самых разных уровнях – от отдельных серверов и кластеров до суперкомпьютеров. Применение в современных процессорах x86 большого числа ядер и сдвигает серверы с x86 до уровня возможного эффективного применения облачной технологии.

В обзоре облачные технологии практически не рассматриваются, однако для полноты картины полезно привести общую информацию об имеющихся реализациях с применением рассматриваемых в обзоре процессоров, особенно для HPC. Ведь имеются публикации, где данные о производительности и в области HPC для этих процессоров получены при использовании облачной технологии.

Поскольку имеется большое число разных типов облачной технологии, мы упомянем только две, широко используемые для HPC-задач: IaaS (Infrastructure as a Service) и BMaaS (Bare Metal as a Service). Хорошие общие рекомендации по использованию x86-процессоров для разных типов облака имеются для EPYC 9004 в [**221**]. Их можно естественно распространить и на другие современные процессоры.

В качестве примеров использующих такие технологии вычислительных систем, ресурсы которых предоставляются пользователям, укажем предложения Microsoft и Amazon. Microsoft Azure предлагает виртуальные машины с разными процессорами, в том числе и с рассматриваемыми в обзоре масштабируемыми процессорами Xeon 4-го поколения (Xeon 8490H и 8473C) [**361**, **362**], и с EPYC Zen 4 [**363**].

Amazon предлагает и ориентированные на HPC виртуальные машины, в том числе с AMD EPYC Zen 4 (Genoa) [364], которые фирма предлагает использовать, в частности, для задач CFD и предсказания погоды.

Можно также отметить Intel Tiber AI Cloud [365], где в первую очередь для задач ИИ предлагается доступ к самым современным аппаратным средствам фирмы.

Понятно, что в используемых для облачных технологий серверах, особенно для HPC, активно используются и GPU. В качестве завершения данного раздела укажем на применение облачных технологий Oracle: в OCI (Oracle Cloud Infrastructure) анонсированы новые возможности инфраструктуры ИИ для малых, средних и крупных вариантов использования, включая крупнейший суперкомпьютер ИИ в облаке [366]. Там фактически предлагается BMaaS, но вычислительные возможности базируются на GPU [366], и к обзору это уже отношения не имеет.

6. Об Intel Xeon 6 Granite Rapids и AMD EPYC Turin (Zen 5)

Безусловно, появление осенью 2024 года Xeon 6 Granite Rapids с высокопроизводительными P-ядрами (старших моделей, версии AP - Xeon 6900P) и EPYC Zen 5 (моделей 9005) произвело существенное изменение ситуации с серверными процессорами x86-64.

Уже в 2025 году Intel добавила к таким Xeon 6 с P-ядрами версию SP – Xeon 6700P и 6500P с меньшим числом ядер, меньшим числом каналов UPI и меньшим числом каналов памяти [367], а также с другим, более простым сокетом Intel 4710 (в версии AP используется сокет FCLGA7529) [49]. Но у этих процессоров появилась возможность строить конфигурации серверов с числом сокетов до 8. Появилась также серия еще более простых процессоров Xeon 6300P [367].

Другие модели Xeon 6, Sierra Forest, неинтересны для целей данного обзора, поскольку используют более простые E-ядра (не имеющие поддержки AVX-512), поддерживают память с меньшей пропускной способностью и меньшим числом каналов, и имеют меньшую емкость кэша L3. Вариант с E-ядрами в основной на момент написания обзора публикации Intel [368] вообще отнесен к другой микроархитектуре. Мы рассматриваем только Xeon 6 с P-ядрами. Далее под Xeon 6 вообще имеется в виду Granite Rapids с P-ядрами, версии AP.

В процессорах Xeon 6 Intel стала применять свою новую технологию 5 нм. Xeon 6 стал очередной демонстрацией продвижения вперед в направлениях увеличения числа процессорных ядер, доминирующего влияния на производительность именно применяемой полупроводниковой технологии и использования чиплетов.

При не очень большом прогрессе в полупроводниковой технологии Intel переход с Xeon SPR на Xeon EMR не дал фирме возможности обогнать по числу ядер и производительности в широком диапазоне областей применения процессоры EPYC 9004. Переход Intel на технологию 5 нм, несомненно, гораздо более сильный шаг вперед, чем был при переходе с 4-го поколения масштабируемых процессоров Xeon на 5-е поколение.

В таблице 44 представлены только старшие модели процессоров Xeon 6 и EPYC 9005 Turin (с большим числом ядер). Например, всего имеется 26 разных моделей EPYC 9005 с числом ядер от 8, причем ряд отсутствующих в таблице моделей с числом ядер от 48 и ниже имеют TDP 300 Вт и ниже [**126**].

Таблица 44. Основные показатели Xeon 6900P и EPYC 9005

Модель	Число ядер	Частота, ГГц	Емкость кэша L3	TDP, Вт	Цена	Технология
Xeon 6980P[1]	128	2–3.9	480 МБ	500	$12460	5 нм
Xeon 6979P	120	2.1–3.9	432 МБ	500	$11025	5 нм
Xeon 6972P	96	2.4–3.9	480 МБ	500	$10220	5 нм
Xeon 6960P	72	2.7–3.9	504 МБ	500	$9625	5 нм
Xeon 6952P	96	2.1–3.9	504 МБ	400	$9115	5 нм
EPYC 9965[2]	192	2.25–3.7	384 МБ	500	$14813	3 нм
EPYC 9845	160	2.1–3.7	320 МБ	390	$13564	3 нм
EPYC 9755	128	2.7–4.1	512 МБ	500	$12984	4 нм
EPYC 9745	128	2.4–3.7	256 МБ	400	$12141	3 нм
EPYC 9655	96	2.6–4.5	384 МБ	400	$11852	4 нм
EPYC 9565	72	3.15–4.3	384 МБ	400	$10486	4 нм
EPYC 9575F	64	3.3–5	256 МБ	400	$11791	4 нм
EPYC 9475F	48	3.65–4.8	256 МБ	400	$7592	4 нм

[1] Данные для Xeon из [https://ark.intel.com/content/www/us/en/ark/products/series/240357/intel-xeon-6.html 14.10.2024] на 14.10.2024;

[2] Данные для EPYC из [**49**] на 14.10.2024.

Цены для всех процессоров - из [**49**] на 16.04.2025.

В AMD Zen 5 теперь используется технология TSMC 4 нм. По числу процессорных ядер AMD по-прежнему впереди, но это относится только к процессорам на базе ядер Zen 5c (новейший аналог Zen 4c), где применяется технология TSMC 3 нм и уменьшена емкость кэша L3 на ядро по сравнению с ядрами Zen 5, см. таблицу 44. Для уточнения указанных в ней технологий отметим, что в более простых блоках процессоров, например в блоке I/O (IOD), который, хоть и с существенно отличающейся функциональностью, чем в EPYC, теперь имеется и в Xeon 6 (см. [**368**, **369**]), применяются более старые технологии.

Большое увеличение числа процессорных ядер в Xeon 6 потребовало и увеличения пропускной способности межсоединения процессоров (каналов UPI стало 6, со скоростью 24 GT/s – суммарно в 1.8 раз больше, чем в Xeon EMR [**368**]).

Конкурентоспособность старших моделей Xeon 6 относительно EPYC 9005 должна стать существенно более высокой, чем это было у 4-го и 5-го поколения масштабируемых процессоров Xeon относительно EPYC 9004. Xeon 6900P в ряде первоначальных данных о производительности в HPC смогли превзойти EPYC 9005 (см., например, таблицу 45 ниже).

Анализ микроархитектур и производительности требует появления соответствующих данных, которые в настоящее время для Xeon 6 и EPYC 9005 весьма активно пополняются. Но такой анализ в обзоре требует достаточно полных данных, и поэтому здесь не проводится. При этом ясно, что большинство из рассмотренного ранее в данном обзоре остается в силе и в новейших процессорах. Для микроархитектуры EPYC 9005 это видно, например, из документов AMD [**370**, **371**]. В Zen 5 главными приоритетами разработки было увеличение производительности и энергоэффективности на ядро. Информация об усовершенствованиях в ядрах Zen 5 имеется в [**372**, **373**].

Дополнительные к [**373**] данные[48] о докладе AMD на конференции Hot Chips 2024 включают соответствующие слайды. Подробные количественные показатели обоих фронт-энд и бэк-энд компонент микроархитектуры ядер Zen 5 уже представлены в [**374**]. Понятно, что прогресс интегрально выразился в росте IPC (на 16%).

Мы ограничимся далее только краткой информацией о появившихся моделях Xeon 6900P и EPYC 9005, и некоторыми стартовыми данными об их производительности.

Обе фирмы, кроме использования прогресса в технологии, перешли на работу с более быстрой памятью DDR5. В спецификациях процессоров AMD [**126**] указано о поддержке ими до 12 каналов на скорости до 6000 MT/s, хотя в [**200**] отмечается поддержка в некоторых моделях DDR5-6400. Процессоры Xeon 6 поддерживают работу до 12 каналов DDR5-6400 или более высокопроизводительную, хотя и более дорогую память MRDIMM-8800, имеющую более высокую скорость передачи. Влияние повышенной пропускной способности MRDIMM-8800 на производительность по сравнению с использованием DDR5-6400 продемонстрировано в [**375**].

Безусловно, большой прогресс в типе поддерживаемой Xeon 6 памяти должен отразиться на росте достигаемой пропускной способности и соответственно на росте производительности в приложениях механики сплошных сред, включая задачи CFD (это уже стало видно в данных соответствующих PTS-тестов на сайте openbenchmarking.org).

[48] https://chipsandcheese.com/p/discussing-amds-zen-5-at-hot-chips-2024, accessed 26.04.2025.

Переход в Xeon 6 на работу с имеющей более высокую пропускную способность памятью представляется вторым по важности успехом Intel после прогресса из-за новой технологии 5 нм. Однако анализы надо проводить не на пиковых, а на реально достигаемых показателях производительности и пропускной способности памяти. Важны, например, показатели межсоединения ядер в Xeon 6 – поскольку для Xeon Max в [**354**] были отмечены потенциальные ограничения ими пропускной способности памяти.

Данные проведенного Intel теста stream triad для 2P сервера с 96-ядерными Xeon 6972P с памятью MRDIMM-8800 показали преимущество пропускной способности в 1.3 раза по сравнению с 2P сервером также с 96-ядерными EPYC 9655 и памятью DDR5 на скорости 6000 MT/s [**376**]. Кроме того, с Xeon 6 уже продвигаются вперед работы по применению и CXL-памяти [**377**].

Имеются приписываемые Chongqing Microcomputer китайские данные[49] для 2P-сервера с EPYC 9755 по пропускной способности памяти в stream triad, а также по производительности в тестах DGEMM и HPL.

Здесь нужно также отметить, что в Xeon 6 планировалось появление нового векторного расширения ISA, AVX10, с поддержкой ограничения длины векторов в 256 бит. Фирма AMD в EPYC 9005 перешла на применение (при работе с AVX-512) 512-битной ширины передачи данных вместо 256-битной в EPYC 9004. Это не только может существенно повысить достигаемую производительность DGEMM по сравнению с EPYC 9004 (что уже видно в соответствующих PTS-тестах на сайте openbenchmarking.org), но и способствует росту производительности в задачах ИИ [**373**]. Влияние этого фактора на производительность приложений в разных областях применения рассмотрено в [**378**]. Таким образом, AMD относительно Zen 4 продвинулась вперед в работе с векторными операциями, а Intel в Xeon 6 оказалась впереди Zen 5 в используемой технологии памяти.

Intel благодаря новой технологии смогла повысить емкость кэша L3, и в некоторых моделях Xeon 6 в таблице 44 емкость такого кэша на ядро выше, чем в EPYC 9005. AMD, благодаря более продвинутой полупроводниковой технологии, обеспечивает при равном числе ядер более высокие величины тактовых частот.

О процессорах EPYC 9005 большое количество информации, включая данные о производительности в разных приложениях, уже стало доступно на документационном хабе AMD [**146**] среди появившихся в конце 2024 года документов – например, о производительности с OpenFOAM

[49] https://news.qq.com/rain/a/20241202A090KQ00, accessed 6.03.2025.

см. [**379**]. В данных об относительной производительности EPYC 9005 по сравнению с EPYC 9004 и Xeon EMR[50] представлены AMD для четырех CFD-приложений ANSYS. Intel размещает неструктурированные данные о производительности Xeon 6 на своем сайте [**376**].

Стали доступны и данные о производительности EPYC 9005 и Xeon 6 в тестах SPECcpu 2017. Правда, сам смысл этих тестов для современных многоядерных серверных процессоров кардинально понизился. Мы далее основываемся на максимальных достигнутых для рассматриваемых моделей и конкретных тестов официальных результатах сайта на 16.04.2025.

В SPECint 128-ядерный Xeon 6980P отстает от EPYC 9755 с тем же числом ядер, а 96-ядерный Xeon 6972P – от УЗНС 9655 с тем же числом ядер. В SPECfp результат скорее противоположный. В SPECfp_rate, где меряется «пропускная способность» заданий, и в отличие от SPECfp_speed распараллеливание запрещено, 128-ядерный Xeon 6980P достиг 1360/2720 для 1P/2P-конфигураций против аналогичных более низких результатов 1230/2660 для EPYC 9755 с тем же числом ядер. В SPECfp_speed Xeon 6980P достиг 448/496 против 435/573 для EPYC 9755 для 1P/2P-конфигураций.

Для 96-ядерной модели Xeon 6972P в тесте SPECfp_rate достигалось 1160/2290 против 1020/2050 для EPYC 9655 с тем же числом ядер в конфигурации 1P/2P. В SPECfp_speed в 1P-сервере Xeon 6972P максимальная полученная величина составила 394 против 412 для 1P-сервера с EPYC 9655.

Видно, что достигаемые производительности Xeon 6900P и EPYC 9005 при одинаковом числе ядер достаточно близки. Обсуждаться эти результаты здесь не будут. Можно только отметить два момента. Во-первых, большой прогресс в производительности EPYC 9005 по сравнению с EPYC 9004 (сравнивая 96-ядерные модели EPYC 9655 и EPYC 9654 - см. данные в таблице 27); для Xeon 6900P прогресс еще выше. Во-вторых, можно сказать, что в Xeon 6900P отставание от EPYC в тестах SPECcpu было ликвидировано.

В тесте SPECmpi 2007 (medium) 2P-сервер с EPYC 9755 достиг значения 96.6 против 65.4 для 2P-сервера с EPYC 9654 (максимальные показатели на 14.04.2025), что демонстрирует заметно больший рост производительности по сравнению с увеличением числа ядер. Для теста SPEChpc 2021 данные о производительности приведены в таблице 45. Данные для Xeon 6 в этих тестах SPEC отсутствуют.

[50] https://www.amd.com/content/dam/amd/en/documents/epyc-technical-docs/performance-briefs/amd-epyc-9005-pb-ansys-ls-dyna.pdf, accessed 26.04.2025 и аналогичные URL.

Таблица 45. Производительность EPYC 9005 в тестах SPEChpc 2021

Модель	Число/тип ядер	Конфигурация	Размер теста: tiny	Размер теста: small
EPYC 9965	192/Zen 5c	2P	22.9	2.21
EPYC 9755	128/Zen 5	1P	9.24	
		2P	22.1	
EPYC 9555	64/Zen 5	2P	17.7	1.71
EPYC 9754	128/Zen 4c	1P	7.32	
		2P	16.4	
EPYC 9684X	96/Zen 4	2P	16.0	1.55
EPYC 9654	96/Zen 4	1P	6.99	0.735
		2P	14.2	1.59

В таблице приведены только максимальные достигнутые на 14.04.2025 результаты.

К сожалению, данные при одинаковом числе ядер здесь относятся к их разным типам.

Немало данных о производительности EPYC 9005 и Xeon 6980P появилось в PTS-тестах на сайте openbenchmarking.org. На 15.04.2025 в PTS-тестах молекулярной динамики (GROMACS, NAMD, LAMMPS) Xeon 6980P чаще совсем немного отставал по производительности от старших моделей EPYC 9005. Отставание также наблюдалось в приложениях молекулярного докинга (miniBUDE) и квантовой химии (NWChem).

В одних PTS-тестах из числа входящих в NPB немного быстрее был Xeon 6980P, в других – EPYC 9755. В этих данных PTS-тестов надо обращать особое внимание на «эффективность масштабирования» производительности с числом используемых ядер, про что многократно говорилось выше.

Из данных о производительности вывода моделей ИИ укажем на PTS-тесты на базе созданных Intel средств OpenVINO, в которых (на 17.01.2025) процессоры Xeon 6980P иногда опережали EPYC 9755, а иногда отставали. Автор не располагает данными об использовании AMX или оптимизированных программных средств Intel в этих тестах. Мы далее ограничимся только сравнительными данными из [**200**].

В таблице 46 приведено очень небольшое количество отобранных данных из полученных в [**200**] результатов. Отбор был ориентирован на область HPC, хотя в таблице есть и данные для теста TensorFlow. Проведенный выбор не может компенсировать указанные выше недостатки некоторых PTS-тестов, хотя менее информативные результаты мы старались не приводить.

Таблица 46. Данные о производительности процессоров Xeon 6, EPYC 9005 и предыдущих поколений Xeon и EPYC в разных тестах производительности

Модель/ число ядер	n	HPCG 3.1, GFLOPS	NPB 3.4 MG.C, MOPS[2]	Open FOAM medium, секунд	NWChem, секунд	GROMACS, нс/день	TensorFlow ResNet-50, BS=512, изображений/с
Xeon 6980P/ 128 (с MRDIMM)[1]	1	86.32	240025.24	123.73	1562.7	20.545	234.72
	2	170.16	451798.00	73.45	1609.5	32.669	180.57
Xeon 6980P/ 128	1	65.22	196892.69	131.37	1564.2	21.086	223.12
	2	127.62	387445.06	71.30	1612.6	33.120	178.53
Xeon 8592+/ 64	1	35.14	107321.89	302.06	1878.3	10.477	140.53
	2	69.30	220893.78	137.04	1809.7	18.540	169.75
Xeon Max 9480/56	1	30.59	86407.46	410.29		8.282	125.14
	2	62.64	171436.47	186.33		14.127	138.39
Xeon 8490H/ 60	1	31.07	83448.70	420.65		8.762	148.45
	2	60.85	163333.21	196.50		18.540	174.95
EPYC 9965/ 192	1	60.70	112588.99	176.40	1557.9	23.142	247.22
	2	93.62	223715.54	79.38	1594.0	30.707	204.43
EPYC 9755/ 128	1	63.67	167026.93	160.37	1326.2	22.729	246.24
	2	64.02	361767.36	74.46	1310.1	34.382	210.75
EPYC 9575F/ 64	1	63.90	159641.78	233.12	1225.7	14.651	254.02
	2	101.90	301867.19	103.87	1127.7	26.665	279.74
EPYC 9754/ 128	1	23.63	125567.07	449.47	1700.0	13.242	135.17
	2	41.87	245447.72	137.11	1712.6	21.305	174.93
EPYC 9684X/ 96	1	27.58	134994.14	178.25	1450.5	15.222	121.96
	2	44.95	244738.81	93.57	1429.9	22.525	137.50
EPYC 9654/ 96	1	28.32	116725.04	325.42	1575.5	12.514	161.90
	2	42.52	194942.32	137.11	1558.4	20.234	133.87

Примечание. Красным помечены наиболее высокопроизводительные результаты тестов (отдельно для 1P- или 2P-конфигураций). Это также показывает, большие или меньшие величины отвечают большей производительности в конкретном тексте. Синим цветом помечены данные для процессоров на ядрах Zen 5c или Zen 4c. Все варианты тестов и результаты тестов в данной таблице – из соответствующих результатов PTS-тестов на сайте openbenchmarking.org и в [200]. Используемые в таблице величины: n – число процессоров, BS – размер пакета.

[1] В случае отсутствия указания на используемую память в Xeon 6 подразумевается работа с DDR5-6400.

[2] NPB MG.C выбран из семейства тестов NPB 1.4 по соображениям большей отдаленности от области HPCG. В тестах SP.C и EP.D быстрее был Xeon 6, в тестах CG.C, LU.C и IS.D – EPYC 9005.

В этой таблице представлены результаты [200] для тестов HPCG (с размерами подсеток 144) и OpenFOAM (drivaerFastback, Medium Mesh Size), где пропускная способность памяти должна быть существенной для максимизации производительности, NPB MG.C, а также двух других обычно относимых скорее к вычислительно-интенсивным приложений HPC из области вычислительной химии – NWChem (квантовая химия) и GROMACS (молекулярная динамика).

Кроме того, в таблице представлены данные из [**200**] для известного PTS-теста для задач вывода ИИ - в модели RESnet-50 для фреймворка TensorFlow для пакетов размером 512. Здесь надо иметь в виду, что производительность такого теста слабо масштабируется с увеличением числа ядер при замене модели ЦП, при переходе от 1P- к 2P-конфигурациям с EPYC 9004 и Xeon EMR/SPR. Более того, на новейших процессорах Xeon 6 и EPYC Zen 5 производительность вообще уменьшалась при переходе от 1P- к 2P-конфигурации. Эти данные приведены в таблице и для иллюстрации того, что для широко распространенных тестов часто требуется аккуратный анализ для получения правильных выводов.

Несмотря на очень небольшой период после появления Xeon 6 и EPYC 9005, приведенные выше данные все-таки дают достаточно широкий охват областей применения для самого первоначального уровня оценок.

Хотя в таблице 46 часть тестов продемонстрировали преимущества производительности EPYC 9005, а другая – Xeon 6, не следует воспринимать это как имеющее общий характер. В [**200**], как и в результатах PTS-тестов на сайте openbenchmarking.org, уже имеется огромное количество других данных. Важно только аккуратно смотреть, какой смысл имеют эти результаты.

Но в целом, в отличие от сопоставленных в обзоре EPYC 9004 и Xeon SPR/EMR, процессоры Xeon 6 с момента своего появления демонстрирует, в том числе в области HPC, заметно большее число преимуществ производительности относительно EPYC 9005, чем это было в области «вокруг четвертого поколения» серверных процессоров x86.

Утверждение в [**200**] о доминировании производительности EPYC 9005 можно отнести только к использованному там набору тестов; в общем случае такое утверждение выглядит преждевременным. Надо также иметь в виду, что более высокая производительность могла быть достигнута там на процессорах с меньшим числом ядер.

Из сказанного выше о производительности становится еще более важным учет стоимости и энергопотребления. Цены для Xeon 6900P и EPYC 9005 на момент написания обзора приведены в таблице 44. Надо отметить, что ранее цены Xeon 6900P разработчиком предлагались существенно более высокие, заметно превосходящие цены аналогов (в первую очередь по числу ядер) в EPYC 9005, а в 2025 году Intel понизила цены [**367**]. Хотя сегодняшние цены у аналогичных процессоров Xeon и

EPYC близки, у Xeon они чуть ниже. Указанные TDP у EPYC 9755 и Xeon 6980P совпадают. Но по данным [**200**], в применявшихся там тестах средние и пиковые значения использованных процессорами EPYC 9965, 9755 и 9575F мощностей были заметно ниже, чем у Xeon 6980P, который имел пиковое энергопотребление 547 Вт, заметно выше своего TDP.

В целом, что касается сопоставления потенциальной эффективности применения Xeon 6 и AMD EPYC Turin, то, хотя появилось уже немало данных о производительности вычислительных систем на их основе, информации для взвешенного их анализа в настоящее время все еще недостаточно.

7. О построении узлов кластеров и суперкомпьютеров с использованием серверных x86-процессоров

В этом разделе с учетом проведенного выше анализа сформулированы некоторые общие подходы для использования серверных процессоров x86 при построении кластеров и суперкомпьютеров для задач HPC и ИИ, и даны ссылки на соответствующие публикации.

Для таких вычислительных систем сегодня возникает 3 разных базовых варианта.

(*1*) Построение на базе гомогенных узлов

(*2*) Гибридные варианты, когда один подкластер строится из гомогенных узлов, а другой содержит GPU (пример такого – суперкомпьютер LUMI[51])

(*3*) Построение с использованием гетерогенных узлов, содержащих процессоры x86 и GPU

Оптимальный выбор процессоров для вариантов (*1*) или (*3*), как и для соответствующих частей кластеров для второго варианта, может быть разный.

По умолчанию в этом разделе предполагается стремление к максимизации производительности. В случае ориентации на стоимость, отношение производительность/стоимость, энергоэффективность или плотность упаковки об этом явно говорится.

[51] https://docs.lumi-supercomputer.eu/hardware/, accessed 18.04.2025.

Кластеры и суперкомпьютеры с гомогенными узлами. Классическим примером использования кластеров с гомогенными узлами, ориентированным на задачи обработки больших данных (включая работу с БД) и ИИ, является применение кластеров Apache Spark и Hadoop (см., например, [**380**–**385**]). Однако в работе с ними можно использовать и GPU (см., например, [**386**–**389**]). Хотя целесообразность (учитывая и стоимостные показатели) применения здесь GPU в некоторых случаях, естественно, вызывает вопросы [**390**].

Далее в основном предполагается ориентация вычислительных систем на HPC. В случае кластера из гомогенных узлов, естественно, оптимальнее такие процессоры x86, которые и дают максимальную производительность узла там, где он будет использоваться. Для кластера или суперкомпьютера, ориентированного на использование в определенных заранее известных областях применения, может оказаться выгодным, например, отказ от необходимости поддержки высокопроизводительной реализации AVX-512, обеспечиваемой масштабируемыми процессорами Xeon 4-го и 5-го поколений, что соответственно дает потенциальные преимущества EPYC Zen 4.

В случае с небольшими кластерами, вероятно, стоимость процессоров более важна. Соответственно более естественной может быть ориентация на процессоры с более высоким соотношением производительность/стоимость, которое может быть лучше для процессоров среднего класса [**100**]. Полезная общая ориентация для построения экономически эффективных кластеров с узлами на базе рассматриваемых в данном обзоре процессоров для обработки больших данных дана в [**391**]. Для задач CFD имеется, например, специальная методика выбора аппаратных средств с учетом стоимости и производительности [**392**].

Но надо иметь в виду возможности эффективного распараллеливания используемых приложений внутри каждого узла, что важно в первую очередь для таких кластеров, где достаточно четко определена область их применения. При очень большом числе ядер в процессоре может возникать ситуация, когда из-за ограниченности пропускной способности памяти распараллеливание с использованием свыше определенного числа ядер процессора может стать бессмысленно, поскольку производительность далее практически не растет или начинает уменьшаться. Это может возникать у связанных памятью приложений, если пропускная способность памяти насыщается еще до использования всех доступных в процессоре ядер. Конечно, возникновение ситуации с невыгодностью увеличения числа ядер в процессорах зависит от используемого теста или приложения и размера задачи.

Плохое масштабирование производительности с увеличением числа используемых ядер в сервере чаще встречается и наиболее известно в области CFD. Так, в [**393**] изучена производительность при использовании релаксационных схем Рунге-Кутта в 2P-сервере с 16-ядерными Xeon E5-2698v3, и ухудшение роста производительности слабо зависело от размера задачи. Подобное ярко проявлялось еще в период использования многоядерных Intel Xeon с архитектурой MIC (Many Integrated Core) – см., например, [**394**].

Естественно, в CFD имеются противоположные данные о росте производительности при переходе к работе с содержащим большее число ядер и более дорогим процессором, для EPYC 9004 – см., например, данные PTS-тестов OpenFOAM и WRF в [**147**].

Но производительность CFD-приложения с увеличением числа ядер процессора после определенного порога может расти существенно ниже линейного ускорения. В тех же тестах в [**147**] для старших моделей EPYC 9004 прирост производительности в WRF был очень мал, а по соотношению производительность/стоимость 64-ядерные EPYC 9554 обгоняли 96-ядерные EPYC 9654.

Очевидно, что ухудшение масштабирования производительности только иногда обусловлено недостатками пропускной способности памяти, хотя на работу с памятью в таких случаях внимание обращают всегда (см., например, [**395**]).

Подобные ситуации возникают и при работе с DGEMM. Так, в [**396**] для DGEMM из MKL рост производительности 2P-сервера с 24-ядерными Xeon 8160 (Skylake) прекращался при использовании 16 ядер, далее производительность падала. В [**397**] производительность DGEMM из разработанных Intel средств Parallel Research Kernels в 2P-сервере с 8-ядерными Xeon E5-2450 прекращала расти при 8 ядрах.

Аналогичные ситуации в кластерах носят более сложный характер. Появился также специальный термин недоподписка (undersubscription), который используется в случаях, когда в узлах кластера для достижения большей производительности имеет смысл использовать в расчете меньшее число ядер, чем имеется в наличии, а причина этого до сих пор не выявлена. Естественно, такое наблюдается в CFD. Так, было показано, что в приложении Ansys Fluent за счет недоподписки при работе с 60-ядерными EPYC Zen 3 (7V73X) можно получить ускорение расчета до 50% [**395**]. Возможно, эффект недоподписки имеет место в случае распараллеливания с применением турбо-частот процессоров в узлах кластера.

Использование в узлах процессоров с меньшим числом ядер дает не только заметно меньшую стоимость, обычно более важную для кластера, но и возможно определенный рост производительности за счет более высоких тактовых частот в таких процессорах, что характерно для моделей EPYC 9004. Аналогично может оказаться, например, эффективнее применение в узлах Xeon Gold, а не Xeon Platinum. Это соответствует сказанному чуть выше о возможной ориентации на модели процессоров среднего класса. В предельном случае, когда потребность в достигаемом уровне производительности не так высока, может оказаться, что по соотношению производительность/стоимость выгоднее создать кластер из двух узлов с процессорами с меньшим числом ядер, чем использовать один сервер с процессорами с большим числом ядер.

С другой стороны, с увеличением числа узлов в кластере также может наблюдаться отклонение масштабирования производительности в сильно худшую от ее линейного роста сторону. Для определения, сколько узлов в кластере и какие узлы/процессоры оптимально выбрать для решения задач CFD, в [**398**] предложена общая методика. В ней учитывается производительность и стоимость (и соответственно энергоэффективность). Хотя эта методика была продемонстрирована на примере приложения Fluent, она применима и для других приложений.

Для больших кластеров и суперкомпьютеров их четкая ориентация на определенные приложения мало вероятна, скорее предполагается применение самых разных, заранее непредсказуемых приложений. Соответственно оптимизация выбора процессоров для определенных приложений бессмыслена. Для суперкомпьютеров чаще выбираются топ-модели соответствующих семейств процессоров, и больше важны энергоэффективность (и соответственно важна TDP) и плотность упаковки. Соответствующие вопросы практичного и эффективного построения узлов суперкомпьютеров с производительностью свыше 100 TFLOPS (в том числе с ориентацией на экзамасштабируемые суперкомпьютеры и применение GPU) рассмотрены в [**399**].

Кластеры и суперкомпьютеры с гетерогенными узлами. Для случаев построения вычислительной системы с гетерогенными, содержащими процессоры x86 и GPU узлами, ситуация в основном иная.

В гетерогенных узлах с GPU требования к процессорам во многом иные, чем для гомогенных узлов. Задачи процессоров x86 – загружать и

обслуживать ОС, запускать на выполнение программы, и самое важное - производить обмен данными с одним или несколькими GPU в узле. В «предельном случае» практически все обеспечение производительности передается в GPU, задача процессора - в основном коммуникации с GPU (или - не обязательно - между узлами). Все расчеты по сути выполняются на GPU. Тогда главным становится обеспечение быстрого обмена данными с GPU, а не высокая производительность самих процессоров.

Хотя все суперкомпьютеры требуют максимизации энергоэффективности и плотности упаковки, для содержащих GPU узлов (а применение GPU повышает энергоэффективность) это может стать еще более важным. Поэтому считающиеся в первую очередь энергоэффективными процессоры ARM здесь становятся весьма актуальными.

Сначала рассмотрим вариант, когда процессоры и GPU в узлах отделены друг от друга, а потом – вариант с интегрированными в единую микросхему процессорами с GPU.

Как указано в [**391**], при использовании прямого удаленного доступа к памяти (RDMA) применение IOD в EPYC 9004 с интеграцией там контроллера памяти и PCIe дает короткий путь между памятью и PCIe, что способствует уменьшению задержки и увеличению использования пропускной способности обменов данных между памятью и PCIe. Это важно для вычислительных систем, содержащих GPU, которые часто подсоединяются по PCIe. В Xeon EMR обращение к памяти и PCIe распределено между ядрами, что увеличивает длину пути таких передач данных. Но в других ситуациях это дает плюс для Xeon EMR [**391**]. Здесь следует отметить, что в Xeon 6 также стал применяться IOD (см. раздел 7 выше).

Все это становится более важным, поскольку в современных суперкомпьютерах увеличивается число GPU в узле, приходящихся на один процессор [**144**]. Так, в суперкомпьютере Titan это соотношение было 1:1, в Summit – 3:1, а в Frontier – 4:1 [**399**]. И во Frontier более 99% операций с плавающей запятой приходится на графические процессоры [**144**].

Соответственно становится более важной и максимизация пропускной способности оперативной памяти с процессором, для чего необходимо использовать оптимальную NUMA-конфигурацию. Так, в 1P-узлах суперкомпьютера Frontier с 64-ядерными EPYC 7A53 используется NPS=4. Пропускная способность в тестах stream с NPS=4 достигала 180 ГБ/с, а с NPS=1 – только около 125 ГБ/с [**144**].

Также важнее становится поддержка в процессоре возможно специфического межсоединения GPU производителя, или универсальных PCIe или CLX. Применение для этих целей ARM-процессоров может оказаться более эффективным, что и было реализовано в объединенной микросхеме Nvidia GH200 [**2**].

Но на сегодня подавляющее большинство серверов с GPU используют x86-процессоры Xeon или EPYC. Возможно пониженные требования к производительности многоядерных x86 в гетерогенных узлах, даже содержащих по несколько GPU, приводят к тому, что в серверах-узлах суперкомпьютеров часто оказывается достаточным применение одного процессора с большим числом ядер, а не работа с традиционными для HPC 2P-серверами.

Иллюстрацией этого являются, например, мощные суперкомпьютеры ноябрьского списка TOP500 2024 года, которые используют в узлах один оптимизированный 64-ядерный EPYC Trento Zen 3 с четырьмя GPU AMD MI250X – это бывший лидер списка Frontier [**144**], занимающий 5-е место в этом списке лидирующий по производительности в Европе суперкомпьютер HPC6[52] и находящийся на 8-м месте LUMI G[53]. Занимающий там 19-е место Perlmutter[54] содержит в узлах один 56-ядерный EPYC 7763 с четырьмя GPU Nvidia A100. В этих суперкомпьютерах используются современные GPU как от AMD, так и от Nvidia. Но одного процессора с меньшим числом ядер в узле может быть недостаточно.

Можно указать также на входящий в TOP500 немецкий суперкомпьютер HoreKa-Teal, где в узлах совместно с четырьмя GPU Nividia H100 применяются по два 32-ядерных EPYC 9354 [**145**].

Здесь приводились примеры с процессорами AMD EPYC, поскольку они активно используются в лидирующих суперкомпьютерах TOP500 совместно как с GPU от AMD, так и с GPU от Nvidia. Это во многом связано и с активным применением в суперкомпьютерах готовых узлов HPE Cray, например, EX235a (с GPU AMD) и EX235n (с GPU Nvidia) [**403**]. В кратком описании этих узлов[55] указывается только количество сокетов и тип процессора, а не его конкретная модель.

---

[52] https://www.eni.com/en-IT/actions/energy-transition-technologies/supercomputing-artificial-intelligence/supercomputer.html, accessed 20.04.2025.

[53] https://docs.lumi-supercomputer.eu/hardware/lumig/, accessed 18.04.2025.

[54] https://docs.nersc.gov/systems/perlmutter/architecture/, accessed 18.04.2025.

[55] https://www.hpe.com/psnow/doc/a00094635enw?jumpid=in_pdfviewer-psnow, accessed 19.04.2025.

Процессоры Xeon 8480C, содержащие по 56 ядер, также часто используются в узлах суперкомпьютеров с GPU Nvidia H100. В суперкомпьютере Aurora (второе место в списке TOP500) в узлах применяется по два 52-ядерных Xeon Max 9470C и по 6 GPU от Intel (ранее именовавшихся Ponte Vecchio) [**353**]. Для уточнения отметим, что модели с суффиксом C отличаются от «обычных» только несколько пониженными максимальными температурными показателями и отсутствием некоторых неинтересных для задач обзора опций.

Здесь также нужно обратить внимание на то, что все больший процент серверов с GPU или гетерогенных узлов, содержащих GPU, используются для задач ИИ. Там для наиболее крупномасштабных таких задач требуется огромная емкость памяти, и в «GPU-части» также. Но соответственно процессоры серверов/узлов должны иметь возможность работы с очень большим адресным пространством и в физическом плане также, и ожидаемое распространение CXL-памяти, поддержка которой имеется в рассматриваемых в обзоре процессорах AMD и Intel, может стать большим потенциальным плюсом.

Использование в узлах процессоров, интегрированных с GPU, снимает вопрос выбора процессоров для узла, поскольку это уже проделано производителем. В ряде суперкомпьютеров ноябрьского списка TOP500 2025 года в узлах применяется Nvidia GH200, где с GPU интегрирован 72-ядерный ARM Grace [**13**, **357**] (наивысшее среди них, седьмое место занимает суперкомпьютер Alps [**404**]). Но не менее активно в этом списке использовался и AMD MI300A, в котором с GPU интегрирован 24-ядерный процессор Zen 4 [**405**]. Лидер TOP500, экзафлопсный суперкомпьютер Ливерморской национальной лаборатории имени Лоуренса El Capitan, содержит в узлах по четыре MI300A. В этих суперкомпьютерах также применяются готовые узлы HPE Cray EX254n и EX255a [**403**].

Выводы

Приводимые далее выводы относятся к рассматриваемому в обзоре условному четвертому поколению Xeon и EPYC, если явно не обговаривается иное (например, для процессоров Xeon 6 или EPYC Zen 5).

1. Современное состояние серверных процессоров x86-64 показывает, что в целом преимущество в производительности и энергоэффективности (а также в стоимостном плане) сегодня получает фирма, имеющая преимущество в используемой полупроводниковой технологии (AMD с технологией от TSMC поэтому чаще опережала Intel). Это подтверждается и успехом Intel с Xeon 6 после появления новой технологии Intel 5 нм.
2. В случае применения современных полупроводниковых технологий от 10 нм (Intel Xeon SPR) и ниже оптимальным для таких процессоров становится применение чиплетов и, возможно, трехмерных технологий.
3. Современные ARM-процессоры по производительности (и производительности на ядро) продолжают отставать от старших моделей процессоров x86, отнесенных в обзоре к условному 4-му поколению. Хотя преимуществами ARM-процессоров по сравнению с серверными процессорами x86-64 обычно считают энергоэффективность, данные тестов SPECpower_ssj 2008 показывали противоположные результаты.
4. Хотя в большинстве широко распространенных областей применения серверных процессоров EPYC 9004 опережают по эффективности Xeon SPR (включая Xeon Max) и Xeon EMR, имеется много важных более узких ниш, в которых эти поколения Intel Xeon могут оказаться эффективнее.

   В области HPC это могли бы быть Xeon EMR с приложениями, где производительность лимитируется умножениями матриц. Хотя это относится к достаточно небольшой части HPC, а прямые данные о более высокой производительности DGEMM для Xeon 8592+ по сравнению со старшими моделями EPYC 9004 автору неизвестны. Большее число ядер в таких моделях EPYC элиминирует возможные плюсы Xeon EMR. А, например, в активно использующем умножения матриц тесте HPL имеются приведенные выше данные о более высокой производительности EPYC.

   В области ИИ (если для соответствующей задачи не лучше сразу применять GPU) Xeon (особенно Xeon EMR) может быть эффективнее благодаря применению AMX-расширения в ISA.

   В области применения БД в памяти преимущества Xeon вызваны поддержкой до восьми сокетов в серверах, что дает возможность работать с общей памятью очень большой емкости, которая недоступна для 2P-серверов. Поддержка работы с серверами на базе Xeon, имеющими свыше «обычных» двух сокетов является общим преимуществом, которое может проявляться и в других областях применения, например, и для задач ИИ.

Применение в Xeon специализированных акселераторов создают другие возможности для образования таких ниш.

5. Хотя преимущества серверных процессоров AMD стали возникать достаточно давно (после появления процессоров AMD Zen 2 Rome в 2019 году), с EPYC 9004 область их оптимального применения (вплоть до момента появления Intel Xeon 6 Granite Rapids) максимально расширилась.

   С появлением осенью 2024 года процессоров Intel Xeon 6 Granite Rapids (Xeon 6900P) и AMD Zen 5 Turin (EPYC 9005) с новыми полупроводниковыми технологиями ситуация поменяется. Хотя во многих случаях процессоры EPYC 9005 показывают более высокую производительность, Xeon 6900P демонстрирует возможную ликвидацию достаточно долговременного отставания серверных процессоров Intel по производительности. В Xeon 6900P Intel добилась успеха и за счет увеличенной пропускной способностью памяти (где раньше были отставания от AMD по производительности в связанных памятью приложениях), а AMD сделала определенный шаг вперед при работе с AVX-512, где она могла отставать в производительности.

6. По мере роста числа ядер в процессорах стали чаще возникать ситуации, когда для традиционных HPC-приложений применение более старших и более дорогих моделей с очень большим числом ядер в конкретных расчетах может дать более низкую производительность, чем применение процессоров с меньшим числом ядер (и с повышенными тактовыми частотами). Ранее такое было известно в основном для задач CFD – когда эффективнее было применять кластеры с процессорами, имеющими меньшее число ядер.

   В старших моделях с большим числом ядер может оказываться более низким показатель теоретической пропускной способной способности памяти в расчете на одно ядро, а имеющиеся данные, например, для Xeon SPR и Xeon Max, говорят о существенном уменьшении прироста пропускной способности памяти при увеличении числа ядер. Все это соответственно способствует ухудшению распараллеливания с ростом числа ядер.

   Это привело к недостаточности данных от широко выполняемых тестов производительности (в том числе с применением приложений), поскольку в них обычно отсутствует анализ зависимости производительности от числа применяемых ядер (например, в тестах PTS в openbenchmarking.org). В результате некоторые оценки производительности во многом теряют свой смысл и не демонстрируют эффективности применения различных моделей процессоров.

7. Применение аппаратных средств поддержки безопасности в рассматриваемых процессорах, возможно, будет расширяться по мере расширения использования облачных технологий. Созданная Intel новая технология TDX была основана на огромном объеме работы, в том числе в сотрудничестве с другими известными компаниями. TDX включает расширения в ISA и требует доработки приложений для ее применения. Поэтому TDX – это большой задел на будущее, в первую очередь для облачных приложений. AMD в EPYC использует чисто аппаратные средства безопасности, более простые по сути.

   Однако данные о том, что применение средств безопасности в современных процессорах вызывает большие понижения производительности для HPC, а также необходимость применения для работы с TDX модернизации кодов приложений делают это неэффективным для высокопроизводительных вычислений.

8. Важнейшим потенциальным преимуществом Intel остаются созданные фирмой средства разработки программ, включая oneAPI с DPC++. Несмотря на наблюдавшиеся преимущества производительности современных серверных процессоров AMD, традиционное широкое применение SDK Intel может способствовать сохранению ориентации на Xeon. Хотя применение oneAPI как открытого стандарта с открытым исходным кодом может отчасти одновременно и нивелировать плюсы Xeon по отношению к EPYC.

9. Возможно, применение современных серверных процессоров для задач ИИ будет расширяться, в том числе и с применением кластеров Apache Hadoop или Spark. В первую очередь это вероятно по отношению к выводам моделей ИИ. Но в каких случаях и насколько это может быть реально эффективно по сравнению с работой на GPU, пока не совсем ясно.

10. Что касается суперкомпьютерных технологий и построения кластеров, сказанное выше ранее давало преимущества для EPYC при их использовании как в гомогенных узлах, так и в гетерогенных с использованием GPU. Для гомогенных узлов без GPU появление Xeon 6 дает возможное нивелирование отставания в производительности этого поколения Xeon по сравнению с Zen 5. В гетерогенных узлах с GPU могут быть несколько иные задачи для процессоров, с большими требованиями к пропускной способности памяти, которая может оказаться в настоящее время выше в Xeon 6 с более высокоскоростной памятью. Важным может стать и преимущественная ориентация AMD на задачи HPC, хотя EPYC 9004 активно применяются и для задач ИИ.

Список использованных источников

[1] *Top500 the list*, The Top500 ranking, 63rd.– 2025. URL ↑280, 290

[2] Кузьминский М. Б. *Новое поколение GPGPU и дополнительных аппаратных средств для построения вычислительных систем с этими графическими процессорами: микроархитектура и производительность от серверов до суперкомпьютеров* // Программные системы: теория и приложения.– 2024.– Т. **15**.– № 2(61).– С. 139–473. URL doi ↑280, 281, 285, 301, 309, 317, 404, 480

[3] Drávai B., Reguly I. Z. *Benchmarking the evolution of performance and energy efficiency across recent generations of Intel Xeon processors* // *PMBS 2024: 15th IEEE International Workshop on Performance Modeling, Benchmarking, and Simulation of High Performance Computer Systems* (Atlanta, GA, USA, 17–22 November 2024).– IEEE.– 2024.– ISBN 979-8-3503-5554-3.– Pp. 1413–1419. doi ↑280, 453, 462

[4] Torres L. A., Barrios C. J., Denneulin Y. *Evaluation of computational and energy performance in matrix multiplication algorithms on CPU and GPU using MKL, cuBLAS and SYCL*.– 2024.– 14 pp. arXiv 2405.17322 ↑280, 426

[5] *Top500 the list*, The Top500 ranking, 61sd.– 2023. URL ↑280

[6] Duan X., Wang J., Gao P., Ma M., Gan L., Liu X., Fu H., Xue W., Chen D., Yang G., Liu W. *Enabling real world scale structural superlubricity all-atom simulation on the next-generation sunway supercomputer* // *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Denver, CO, USA, 12–17 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0109-2.– id. 99.– 14 pp. doi ↑280

[7] Кузьминский М. Б. *Современные серверные ARM-процессоры для суперЭВМ: A64FX и другие. Начальные данные тестов производительности* // Программные системы: теория и приложения.– 2022.– Т. **13**.– № 1 (52).– С. 63–129; Kuzminsky M. B. *Modern server ARM processors for supercomputers: A64FX and others. Initial data of benchmarks* // Program Systems: Theory and Applications.– 2022.– Vol. **13**.– No. 1(52).– Pp. 131–194. URL doi URL doi ↑280, 306, 404

[8] Simakov N. A., Deleon R. L., White J. P., Jones M. D., Furlani T. R., Siegmann E., Harrison R. J. *Are we ready for broader adoption of ARM in the HPC community: Performance and energy efficiency analysis of benchmarks and applications executed on high-end ARM systems* // *International Conference on High Performance Computing in Asia-Pacific Region Workshops*, HPCASIAWORKSHOP 2023 (Raffles Blvd, Singapore, February 27–March 2, 2023), New York: ACM.– 2023.– ISBN 978-1-4503-9989-0.– Pp. 78–86. doi ↑280, 288, 289

[9] Rahman T. N., Khan N., Zaman Z. I. *Redefining computing: Rise of ARM from consumer to Cloud for energy efficiency*.– 2024.– 19 pp. arXiv 2402.02527 ↑280

[10] Loghin D. *Are ARM cloud servers ready for database workloads? An experimental study* // IEEE Transactions on Cloud Computing.– 2024.– Vol. **12**.– No. 3.– Pp. 818–829. doi ↑280

[11] Matha R., Kimovski D., Zabrovskiy A., Timmerer C., Prodan R. *Where to encode: A performance analysis of x86 and arm-based Amazon EC2 instances // 2021 IEEE 17th International Conference on eScience (eScience)* (Innsbruck, Austria, 20–23 September 2021).– IEEE.– 2021.– ISBN 978-1-6654-0361-0.– Pp. 118–127. doi ↑280

[12] Larrabel M. *NVIDIA GH200 CPU Performance Benchmarks Against AMD EPYC™ Zen 4 & Intel Xeon Emerald Rapids*, Processors Linux Reviews & Articles.– Phoronix Media.– 2024.– 5 pp. URL ↑281, 348

[13] Laukemann J., Hager G., Wellein G. *Microarchitectural comparison and in-core modeling of state-of-the-art CPUs: Grace, Sapphire Rapids, and Genoa.*– 2024.– 5 pp. arXiv 2409.08108 ↑281, 386, 387, 481

[14] Siegmann E., Harrison R. J., Carlson D., Chheda S., Curtis A., Coskun F., Gonzalez R., Wood D., Simakov N. A. *First impressions of the sapphire rapids processor with HBM for scientific workloads* // SN Computer Science.– 2024.– Vol. **5**.– No. 5.– id. 623.– 11 pp. doi ↑281, 408, 455, 459, 460

[15] Kotra J. B., Kalamatianos J. *Improving the utilization of micro-operation caches in x86 processors // 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)* (Athens, Greece, 17–21 October 2020).– IEEE.– 2020.– ISBN 978-1-7281-7384-9.– Pp. 160–172. doi ↑281

[16] Ren X., Moody L., Taram M., Jordan M., Tullsen D. M., Venkat A. *I see dead μops: Leaking secrets via Intel/AMD micro-op caches // ISCA '21: Proceedings of the 48th Annual International Symposium on Computer Architecture* (Virtual Event, Spain, 14–19 June 2021).– IEEE.– 2021.– ISBN 978-1-4503-9086-6.– Pp. 361–374. doi ↑281

[17] *Introducing Intel Advanced Performance Extensions (Intel APX) Upd. 3/5/2024.*– 2024 (Accessed 31.07.2024(*)). URL ↑281

[18] Blem E., Menon J., Sankaralingam K. *A detailed analysis of contemporary ARM and x86 architectures*, UW-Madison Technical Report.– 2013 (Accessed 12.05.2025).– 13 pp. URL ↑281

[19] Blem E., Menon J., Sankaralingam K. *Power struggles: Revisiting the RISC vs. CISC debate on contemporary ARM and x86 architectures // HPCA '13: Proceedings of the 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA)* (23–27 February 2013).– IEEE.– 2013.– ISBN 978-1-4673-5585-8.– Pp. 1–12. doi ↑281

[20] Blem E., Menon J., Vijayaraghavan T., Sankaralingam K. *ISA wars: Understanding the relevance of ISA being RISC or CISC to performance, power, and energy on modern architectures* // ACM Transactions on Computer Systems (TOCS).– 2015.– Vol. **33**.– No. 1.– id. 3.– 34 pp. doi ↑281

[21] Lam Ch. *Why x86 doesn't need to die.*– 2024 (Accessed 12.05.2025). URL ↑282

[22] Troester K., Bhargava R. *AMD next generation "Zen 4" Core and 4th Gen AMD EPYC™ 9004 server CPU // 2023 IEEE Hot Chips 35 Symposium (HCS)* (Palo Alto, CA, USA, 27–29 August 2023).– IEEE.– 2023.– ISBN 9798350339086.– Pp. 1–25. doi ↑282, 303, 304, 305, 306, 307, 308, 310, 312, 316

[23] *4th Gen AMD EPYC™ processor architecture*, White Paper 3rd Ed.– 2023 (Accessed 2.08.2024). URL ↑282, 294, 305, 306, 307, 309, 312, 313, 315, 317, 319, 320, 324, 325, 378

[24] Polgár P., Menyhárt T., Baksay B. C., Kocsis G., Tajti T. G., Gál Z. *Three level benchmarking of Singularity containers for scientific calculations* // Annales Mathematicae et Informaticae.– 2023.– Vol. **58**.– Pp. 133–146. doi ↑282

[25] Szustak L., Wyrzykowski R., Kuczynski L., Olas T. *Architectural adaptation and performance-energy optimization for CFD application on AMD EPYC™ Rome* // IEEE Transactions on Parallel and Distributed Systems.– 2021.– Vol. **32**.– No. 12.– Pp. 2852–2866. doi ↑282

[26] *Cloud/CPU/Java/Power/Storage/Virtualization Benchmarks* (Accessed 19.12.2024). URL ↑283

[27] *Processors benchmarks results* (Accessed 19.12.2024). URL ↑283

[28] *Official Phoronix Test Suite tests* (Accessed 19.12.2024). URL ↑283

[29] Larabel M. *AMD EPYC™ 7302/7402/7502/7742 Linux Performance Benchmarks*.– 2019 (Accessed 12.05.2025).– 9 pp. URL ↑283

[30] Larabel M. *AMD EPYC™ 7003 "Milan" Linux Benchmarks - Superb Performance*.– 2021 (Accessed 12.05.2025).– 12 pp. URL ↑283

[31] Larabel M. *Intel Xeon Ice Lake vs. AMD EPYC™ Milan Server performance, efficiency & value in*.– 2023 (Accessed 12.05.2025).– 10 pp. URL ↑283

[32] Akter S., Khalil K., Bayoumi M. *A survey on hardware security: Current trends and challenges* // IEEE Access.– 2023.– Vol. **11**.– Pp. 77543–77565. doi ↑283

[33] Francillon A., Castelluccia C. *Code injection attacks on harvard-architecture devices* // *CCS '08: Proceedings of the 15th ACM conference on Computer and communications security* (Alexandria, Virginia, USA, 27–31 October 2008), New York: ACM.– 2008.– ISBN 978-1-59593-810-7.– Pp. 15–26. doi ↑283

[34] Zhang J., Chen C., Cui J., Li K. *Timing side-channel attacks and countermeasures in CPU microarchitectures* // ACM Computing Surveys.– 2024.– Vol. **56**.– No. 7.– id. 178.– 40 pp. doi ↑283

[35] Chen W., Zhao Yu., Zhang Y., Qiang W., Zou D., Jin H. *ReminISCence: trusted monitoring against privileged preemption side-channel attacks* // *Computer Security — ESORICS 2024: 29th European Symposium on Research in Computer Security* (Bydgoszcz, Poland, 16–20 September 2024), Berlin–Heidelberg: Springer-Verlag.– 2024.– ISBN 978-3-031-70902-9.– Pp. 24–44. doi ↑283

[36] Wikner J., Razavi K. *Breaking the barrier: post-barrier spectre attacks* // *2025 IEEE Symposium on Security and Privacy (SP)* (San Francisco, CA, USA, 12–15 May 2025).– IEEE.– ISBN 979-8-3315-2236-0.– Pp. 3516–3533. doi ↑283, 411

[37] Li L., Yavarzadeh H., Tullsen D. *Indirector: high-precision branch target injection attacks exploiting the indirect branch predictor* // *SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), Berkeley: USENIX Ass..– 2024.– ISBN 978-1-939133-44-1.– Pp. 2137–2154.– id. 120. doi ↑283, 411

[38] Aktas E., Cohen C., Eads J., Forshaw J., Wilhelm F. *Intel Trust Domain Extensions (TDX) security review*, Google technical report.– 2023 (Accessed 12.05.2025).– 81 pp. URL ↑283, 400, 401

[39] Wikner J., Trujillo D., Razavi K. *Phantom: exploiting decoder-detectable mispredictions // MICRO '23: Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture* (Toronto, ON, Canada, 28 October 2023–1 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0329-4.– Pp. 49–61. doi ↑284

[40] Trujillo D., Wikner J., Razavi K. *INCEPTION: exposing new attack surfaces with training in transient execution // SEC '23: Proceedings of the 32nd USENIX Conference on Security Symposium* (Anaheim, CA, USA, 9–11 August 2023), Berkeley: USENIX Ass..– 2023.– ISBN 978-1-939133-37-3.– Pp. 7303–7320.– id. 409. doi ↑284

[41] Wang P. L., Paccagnella R., Wahby R. S., Brown F. *Bending microarchitectural weird machines towards practicality // SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), Berkeley: USENIX Ass..– 2024.– ISBN 978-1-939133-44-1.– Pp. 1099–1116.– id. 62. doi ↑284

[42] Jattke P., Wipfli M., Solt F., Marazzi M., Bölcskei M., Razavi K. *ZENHAMMER: Rowhammer attacks on AMD zen-based platforms // SEC '24: Proceedings of the 33rd USENIX Conference on Security Symposium* (Philadelphia, PA, USA, 14–16 August 2024), Berkeley: USENIX Ass..– 2024.– ISBN 978-1-939133-44-1.– Pp. 1615–1633. doi ↑284

[43] Gast S., Juffinger J., Maar L., Royer C., Kogler A., Gruss D. *Remote scheduler contention attacks.*– 2024.– 22 pp. arXiv 2404.07042 ↑284

[44] Kim T., Jang H., Shin Y. *Demoting security via exploitation of cache demote operation in Intel's latest ISA extension.*– 2025.– 18 pp. arXiv 2503.10074 ↑284

[45] *Intel C741 Chipset* (Accessed 28.07.2024(*)). URL ↑285, 402

[46] Pano V., Kuttappa R., Taskin B. *3D NoCs with active interposer for multi-die systems // Proceedings of the 13th IEEE/ACM International Symposium on Networks-on-Chip* (New York, 17–18 October 2019), New York: ACM.– 2019.– ISBN 978-1-4503-6700-4.– Pp. 1–8.– id. 14. doi ↑285

[47] *Second Generation Intel Xeon Scalable Processors Product Brief.*– 2019 (Accessed 12.05.2025(*)). URL ↑285

[48] Atkins B., Chang J., Hatch J., Lee D., Napombejara Ch., Popp M. *The Future of AMD.*– 2007 (Accessed 28.07.2024).– 39 pp. URL ↑286

[49] *CPU Specs Database* (Accessed 17.12.2024). URL ↑286, 310, 314, 338, 364, 375, 378, 391, 419, 451, 467, 468

[50] *Intel Processors Product Specifications* (Accessed 23.02.2025(*)). URL ↑286, 391

[51] *All SPEC CPU2017 Results Published by SPEC* (Accessed 19.05.2025). URL ↑286, 302, 450

[52] *High End CPUs – Intel vs AMD.*– CPU Benchmarks.– 2024 (Accessed 29.07.2024). URL ↑286, 287

[53] McCalpin J. D. *Memory bandwidth and system balance in HPC systems*, SC16 Invited Talk: Memory Bandwidth and System Balance in HPC Systems.– 2016 (Accessed 12.05.2025). URL ↑287

[54] Katz R. H., Hennessy J. L. *High performance microprocessor architectures.*– 1976 (Accessed 19.05.2024).– 17 pp. URL ↑287

[55] Guiang C. S., Milfeld K. F., Purkayastha A., Boisseau J. R. *Memory performance of dual-processor nodes: comparison of Intel Xeon and AMD Opteron memory subsystem architectures // 4th LCI International Conference on Linux Clusters: The HPC Revolution* (San Jose, California, USA, 23–26 June 2003).– 2003 (Accessed 2.05.2025).– 24 pp. URL ↑287

[56] McCalpin J. D. *The evolution of single-core bandwidth in multicore processors.*– 2023; *The evolution of single-core bandwidth in multicore systems — update* (Accessed 21.05.2025). URL URL ↑287

[57] Guest M., Munoz J., Green T. *Performance of community codes on multi-core processors. An analysis of computational chemistry and ocean modelling applications // Computing Insight UK 2022 Conference* (Manchester Central Convention Complex, 1–2 December 2022).– 2022 (Accessed 22.11.2024). URLGuest.pdf ↑288

[58] *HPC Challenge Benchmark.*– 2022 (Accessed 1.12.2024). URL ↑288

[59] Eshelman E. *Detailed specifications of the "Ice Lake SP" Intel Xeon Processor Scalable Family CPUs.*– 2021 (Accessed 12.05.2025). URL ↑289, 290

[60] Eshelman E. *Detailed specifications of the AMD EPYC™ "Milan" CPUs.*– 2021 (Accessed 12.05.2025). URL ↑289, 291

[61] *Top500 the list*, List Statistics of TOP500, 63rd edition.– 2024 (Accessed 12.05.2025). URL ↑290

[62] Nana R., Tadonki C., Dokladal P., Mesri Y. *Energy concerns with HPC systems and applications.*– 2023.– 20 pp. arXiv 2309.08615 ↑292

[63] *Fujitsu server PRIMERGY performance report*, PRIMERGY RX8770 M7 Vers.1.1.– 2023 (Accessed 31.07.2024). URL ↑292, 426, 429, 434

[64] *Calculate the Max Flops on Skylake.*– 2018 (Accessed 8.02.2025). URL ↑293

[65] Mulnix D. L. *Third generation Intel Xeon processor Scalable family on two socket platform technical overview.*– 2024 (Accessed 4.12.2024(*)). URL ↑293

[66] *Intel Xeon 6.*– 2024 (Accessed 4.10.2024(*)). URL ↑293

[67] Karpowicz M. P. *Energy-efficient CPU frequency control for the Linux system //* Concurrency and Computation: Practice and Experience.– 2016.– Vol. **28**.– No. 2.– Pp. 420–437. doi ↑293

[68] Brodowski D., Golde N., Wysocki R. J., Kumar V. *CPU frequency and voltage scaling code in the Linux kernel.*– 2017 (Accessed 12.05.2025). URL ↑293, 295, 296

[69] Sinkar A. A., Wang H., Kim N. S. *Workload-aware voltage regulator optimization for power efficient multi-core processors // 2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (Dresden, Germany, 12–16 March 2012).– IEEE.– 2012.– Pp. 1134–1137. doi ↑293

[70] *Advanced Configuration and Power Interface (ACPI) Specification*, Release 6.5.– UEFI Forum, Inc..– 2022 (Accessed 12.05.2025).– 1126 pp. URL ↑293

[71] Chen J., Wolford R. *Understanding P-state control on Intel Xeon Scalable Processors to maximize energy efficiency.*– 2019 (Accessed 12.05.2025).– 34 pp. URL ↑294

[72] *CPU frequency scaling.*– 2025 (Accessed 8.02.2025). URL ↑294

[73] *OpenSuSE Leap 15.5 Power Management* (Accessed 8.02.2025). URL ↑294, 295, 296

[74] *Chapter 17. Tuning CPU frequency to optimize energy consumption*, A Red Hat training course.– 2024 (Accessed 8.02.2025). URL ↑294, 295, 296

[75] *oneAPI — What is it?*– 2024 (Accessed 11.08.2024(*)). URL ↑297, 300

[76] *High Performance Toolchain: Compilers, Libraries & Profilers*, Tuning Guide AMD EPYC™ 9004. Rev. 1.4.– 2023 (Accessed 5.10.2024).– 54 pp. URL ↑297

[77] *AMD Optimizing C/C++ and Fortran Compilers (AOCC).*– 2024 (Accessed 3.09.2024). URL ↑298

[78] *AMD AOCC User Guide.*– 2024 (Accessed 3.09.2024).– 42 pp. URL ↑298

[79] Larabel M. *AMD AOCC 5.0 Compiler Released With Zen 5 Support, new optimizations.*– 2024 (Accessed 12.10.2024). URL ↑298

[80] *HPE Cray Compiler Environment.*– 2024 (Accessed 12.05.2025). URL ↑298

[81] *HPE Cray Programming Environment User Guide: HPCM on HPE Cray:Supercomputing EX and HPE Cray Supercomputing Systems (24.07)*, S-8023.– 2024 (Accessed 3.10.2024).– 33 pp. URL ↑298

[82] *HPE Cray Supercomputing Programming Environment Software Overview.*– 2024 (Accessed 27.10.2024).– 9 pp. URL ↑298

[83] *NVIDIA HPC SDK Version 24.7 Documentation.*– 2024 (Accessed 3.09.2024). URL ↑298

[84] Saastad O. W., Kapanova K., Markov S., Morales C., Shamakina A., Johnson N., Krishnasamy E., Varrette S. *Best Practice Guide Modern Processors*/ ed. Shoukourian H..– PRACE.– 2020 (Accessed 28.11.2024).– 106 pp. URL ↑299

[85] Halbiniak K., Wyrzykowski R., Szustak L., Kulawik A., Meyer N., Gepner P. *Performance exploration of various C/C++ compilers for AMD EPYC™ processors in numerical modeling of solidification* // Advances in Engineering Software.– 2022.– Vol. **166**.– id. 103078. doi ↑299

[86] Machado R. R. L., Eitzinger J., Laukemann J., Hager G., Köstler H., Wellein G. *MD-Bench: Engineering the in-core performance of short-range molecular dynamics kernels from state-of-the-art simulation packages.*– 2023.– 17 pp. arXiv 2302.14660 ↑299

[87] Dal D., Celik E. *Investigation of the impact of different versions of GCC on various metaheuristic-based solvers for traveling salesman problem* // The Journal of Supercomputing.– 2023.– Vol. **79**.– No. 11.– Pp. 12394–12440. ↑299

[88] *4th Gen AMD EPYC™ Processor Benchmark Report v. 1.0*, CC00204.– 2023 (Accessed 11.10.2024).– 22 pp. ↑299, 302, 306, 339, 346, 347, 363, 364, 365, 366

[89] *EPYC™ CPU Overview.*– 2024 (Accessed 6.10.2024).– 46 pp. Sinica_HPC workshop.pdf ↑299, 330, 331

[90] Wang J., Day E. *LS-DYNA and high-performance computing.*– 2023 (Accessed 27.10.2024).– 20 pp. ↑299

[91] Larabel M. *LLVM Clang 16 vs. GCC 13 Compiler Performance On AMD 4th Gen EPYC™ "Genoa".*– 2023 (Accessed 20.10.2024).– 6 pp. ↑299

[92] *AMD Zen Deep Neural Network (ZenDNN).*– 2024 (Accessed 13.10.2024). ↑300

[93] *AI Inferencing with AMD EPYC™ Processors.*– 2024 (Accessed 09.01.2026).– 11 pp. ↑300

[94] *IFX vs IFORT performance difference.*– 2023 (Accessed 5.09.2024). ↑301

[95] *Intel HPC Toolkit Documentation.*– 2024 (Accessed 12.05.2025(*)). ↑301

[96] *AI Tools Documentation.*– 2024 (Accessed 12.05.2025(*)). ↑301

[97] *Community building blocks for HPC systems* (Accessed 8.09.2024). ↑301

[98] *Intel oneAPI Base Toolkit Documentation.*– 2024 (Accessed 8.09.2024(*)). ↑302

[99] Ruhela D. *Investigating the performance of LLVM-Based Intel Fortran Compiler (ifx)* // *High Performance Computing*, ISC High Performance 2024 International Workshops. ISC High Performance 2023 (Hamburg, Germany, 12–16 May 2024), Lecture Notes in Computer Science.– vol. **15058**, Cham: Springer.– 2025.– ISBN 978-3-031-73715-2.– Pp. 173–184. ↑302

[100] Cuma M. *AMD Genoa and Intel Sapphire Rapids review.*– 2023 (Accessed 13.08.2024).– 12 pp. ↑302, 308, 346, 347, 349, 350, 352, 417, 449, 450, 476

[101] Munger B., Wilcox K., Sniderman J., Tung C., Johnson B., Schreiber R. *"Zen 4": The AMD 5nm 5.7 GHz x86-64 microprocessor core* // *2023 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 19–23 February 2023).– IEEE.– 2023.– ISBN 978-1-6654-9017-7.– Pp. 38–39. ↑303, 304

[102] *Zen 2 — Microarchitectures — AMD* (Accessed 4.10.2024). ↑303

[103] *Zen 3 — Microarchitectures — AMD* (Accessed 4.10.2024). ↑303, 305, 320

[104] *Zen 4 — Microarchitectures — AMD* (Accessed 4.10.2024). ↑303, 304, 305, 320

[105] *AMD "Zen 4" Dies, Transistor-Counts, Cache Sizes and Latencies Detailed.*– 2022 (Accessed 12.05.2025). ↑305

[106] Lam Ch. *AMD's Zen 4 Part 1: Frontend and Execution Engine.*– 2022 (Accessed 4.10.2024). ↑304

[107] Bhargava R., Troester K. *AMD Next Generation "Zen 4" core and 4th Gen AMD EPYC™ server CPUs* // IEEE Micro.– 2024.– Vol. **44**.– No. 3.– Pp. 8–17. ↑305, 312

[108] *AMD Zen 4 13 Percent IPC Uplift Build* (Accessed 21.04.2024). URL ↑305

[109] Domke J., Vatai E., Drozd A., ChenT P., Oyama Y., Zhang L. *Matrix engines for high performance computing: A paragon of performance or grasping at straws?2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (Portland, OR, USA, 17–21 May 2021).– IEEE.– 2021.– ISBN 978-1-6654-1156-1.– Pp. 1056–1065. doi ↑306, 307, 309

[110] *Advanced Vector Extensions 512 (AVX-512) — x86* (Accessed 12.05.2025). URL ↑306, 379

[111] Gottschlag M., Bellosa F. *Mechanism to mitigate AVX-induced frequency reduction.*– 2018.– 12 pp. arXiv:1901.04982 ↑307

[112] Gottschlag M., Brantsch P., Bellosa F. *Automatic core specialization for AVX-512 applications // SYSTOR '20: Proceedings of the 13th ACM International Systems and Storage Conference* (Haifa, Israel, 2–4 June 2020), New York: ACM.– 2020.– ISBN 978-1-4503-7588-7.– Pp. 25–35. doi ↑307

[113] Gottschlag M., Schmidt T., Bellosa F. *AVX overhead profiling: how much does your fast code slow you down?APSys '20: Proceedings of the 11th ACM SIGOPS Asia-Pacific Workshop on Systems* (Tsukuba Japan, 24–25 August 2020), New York: ACM.– 2020.– ISBN 978-1-4503-8069-0.– Pp. 59–66. doi ↑307

[114] Cebrian J. M., Natvig L., Jahre M. *Scalability analysis of AVX-512 extensions* // The Journal of supercomputing.– 2020.– Vol. **76**.– No. 3.– Pp. 2082–2097. doi ↑307

[115] *Ice Lake AVX-512 Downclocking.*– 2020 (Accessed 12.05.2025). URL ↑307

[116] Frumusanu A. *Intel 3rd Gen Xeon Scalable (Ice Lake SP) Review: Generationally Big, Competitively Small.*– 2021 (Accessed 12.05.2025(**)). URL ↑307

[117] *3rd and 4th Gen Xeon Scalable Turbo Frequencies.*– 2023 (Accessed 12.05.2025). URL ↑307

[118] *Accelerate Artificial Intelligence (AI) Workloads with Intel Advanced Matrix Extensions (Intel AMX) ID 785250.*– 2024 (Accessed 25.03.2025(*)). URL ↑309, 440

[119] Burd T., Venkataraman S., Li W., Johnson T., Lee J., Velaga S. *2.2 "Zen 4c": The AMD 5nm area-optimized x86-64 microprocessor core // 2024 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 18–22 February 2024).– IEEE.– 2024.– ISBN 979-8-3503-0621-7.– Pp. 38–40. doi ↑309

[120] *AMD EPYC™ 8004 Series Architecture Overview*, Revision 1.0 (Sept. 2023).– 2023 (Accessed 12.05.2025).– 14 pp. URL ↑310

[121] Mayo K., Rajasekaran S., Pasichnyk I., Kashyap A. *High Performance Computing Tuning Guide for AMD EPYC™ 9004 Series Processors Publication 58002*, Revision 1.5 Jan 2024.– 2024 (Accessed 15.11.2024).– 60 pp. URL ↑310, 313, 315, 319, 320, 321, 322, 323, 324, 328, 340, 341, 342, 345, 346, 347, 348, 360, 361, 386, 428, 431, 460

[122] *AMD EPYC™ 9004 Series Processors Data Sheet.*– 2023 (Accessed 09.01.2026).– 2 pp. URL ↑310, 314

[123] *SLES 15 SP4 Optimizing Linux for AMD EPYC™ 9004 Series Processors with SUSE* (Accessed 12.05.2025(**)). URL ↑310

[124] Mujtaba H. *AMD's Monstrous EPYC™ Genoa CPU For SP5 "LGA 6096" Socket Pictured: Up To 96 Zen 4 Cores, 400W TDP.*– 2022 (Accessed 20.05.2024). URL ↑311

[125] *AMD EPYC™ 8004 Series Processors Data Sheet.*– 2023 (Accessed 12.05.2025).– 2 pp. URL ↑312, 314

[126] *AMD Server Processor Specifications* (Accessed 15.10.2024). URL ↑314, 468, 469

[127] Morgan T. P. *Why AMD "Genoa" Epyc Server CPUs Take The Heavyweight Title.*– 2022 (Accessed 26.05.2024). URL ↑317, 327

[128] *Infinity Fabric (IF) — AMD* (Accessed 3.09.2024). URL ↑317

[129] Lehmann S., Gerfers F. *Channel analysis for a 6.4 Gb/s DDR5 data buffer receiver front-end // 2017 15th IEEE International New Circuits and Systems Conference (NEWCAS)* (Strasbourg, France, 25–28 June 2017).– IEEE.– 2017.– ISBN 9781728110325.– Pp. 109–112. doi ↑320

[130] *Micron's Perspective on Impact of CXL on DRAM Bit Growth Rate.*– 2023 (Accessed 12.05.2025).– 9 pp. URL ↑320, 322

[131] Tang Y., Zhou P., Zhang W., Hu H., Yang Q., Xiang Q., Liu T., Shan J., Huang R., Zhao Ch., Chen Ch., Zhang H., Liu F., Zhang Sh., Ding X., Chen J. *Exploring performance and cost optimization with ASIC-based CXL memory // EuroSys '24: Proceedings of the Nineteenth European Conference on Computer Systems* (Athens, Greece, 22–25 April 2024), New York: ACM.– 2024.– ISBN 979-8-4007-0437-6.– Pp. 818–833. doi ↑320

[132] *5-Level Paging and 5-Level EPT*, White Paper Revision 1.1.– 2017 (Accessed 09.01.2026).– 31 pp. URL ↑324

[133] *AMD Infinity Guard.*– 2024 (Accessed 1.06.2024). URL ↑325

[134] *Processor Family with Full Memory Encryption as a Standard Security Feature.*– 2020 (Accessed 3.08.2024). URL ↑325

[135] *AMD Secure Encrypted Virtualization (SEV).*– 2024 (Accessed 1.06.2024). URL ↑325

[136] Jacob H. N., Werling Ch., Buhren R., Seifert J.-P. *faulTPM: Exposing AMD fTPMs' deepest secrets // 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (Delft, Netherlands, 03–07 July 2023).– IEEE.– 2023.– ISBN 9781665465137.– Pp. 1128–1142. doi ↑325

[137] Wilke L., Wichelmann J., Morbitzer M., Eisenbarth Th. *SEVurity: No security without integrity: Breaking integrity-free memory encryption with minimal assumptions // 2020 IEEE Symposium on Security and Privacy (SP)* (San Francisco, California, USA, 18–21 May 2020).– IEEE.– 2020.– ISBN 978-1-7281-3498-7.– Pp. 1483–1496. doi URL ↑325

[138] Karvandi M. S., Meghdadizanjani S., Arasteh S., Monfared S. Kh., Fallah M. K., Gorgin S., Lee J.-A., van der Kouwe E. *The reversing machine: reconstructing memory assumptions.*– 2024.– 17 pp. arXiv 2405.00298 ↑325

[139] *AMD EPYC™ 4004 Series Processors Data Sheet.*– 2024 (Accessed 1.06.2024).– 2 pp. URL ↑326

[140] *Artificial Intelligence Machine Learning (AI/ML) Tuning Guide for AMD EPYC™ 9004 Series Processors*, Rev.1.1.– 2023 (Accessed 26.11.2024). URL ↑328, 368

[141] *Lenovo ThinkAgile VX655 V3 2U Certified Node (AMD EPYC™ 9004) Product Guide.*– 2024 (Accessed 3.08.2024).– 61 pp. URL ↑328

[142] *Configuring AMD xGMI Links on the Lenovo ThinkSystem SR665 V3 Server.*– 2024 (Accessed 3.08.2024).– 15 pp. URL ↑328, 329

[143] *Compute Nodes.*– 2024 (Accessed 21.08.2024). URL ↑329

[144] Atchley S., Zimmer Ch., Lange J., Bernholdt D., Vergara V. M., Beck Th., Brim M., Budiardja R., Chandrasekaran S., Eisenbach M., Evans Th., Ezell M., Frontiere N., Georgiadou A., Glenski J., Grete Ph., Hamilton S., Holmen J., Huebl A., Jacobson D., Joubert W., Mcmahon K., Merzari E., Moore S., Myers A., Nichols S., Oral S., Papatheodore Th., Perez D., Rogers D. M., Schneider E., Vay J.-L., Yeung P. K. *Frontier: exploring exascale // SC '23: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, New York: ACM.– 2023.– ISBN 979-8-4007-0109-2.– id. 52.– 16 pp. doi ↑329, 479, 480

[145] *HoreKa Hardware Overview.*– 2024 (Accessed 2.08.2024). URL ↑329, 480

[146] *AMD Documentation Hub* (Accessed 11.11.2024). URL ↑330, 470

[147] Larabel M. *AMD EPYC™ 9554 & EPYC™ 9654 Benchmarks — Outstanding Performance For Linux HPC/Servers.*– 2022 (Accessed 20.10.2024).– 15 pp. URL ↑331, 346, 358, 477

[148] Nambiar R. *AMD Ecosystem Poised and Ready for the 4th Gen AMD EPYC™ Processors.*– 2022 (Accessed 13.10.2025). URL ↑331, 345, 346, 375

[149] Larabel M. *AlmaLinux, CentOS Stream, Clear Linux, Debian, Fedora & Ubuntu On AMD 4th Gen EPYC™ Genoa.*– 2023 (Accessed 20.10.2024).– 6 pp. URL ↑331

[150] Larabel M. *AMD 4th Gen EPYC™ 9654 "Genoa" AVX-512 Performance Analysis.*– 2022 (Accessed 11.11.2024).– 9 pp. URL ↑332, 367, 368

[151] Lin W. C., McIntosh-Smith S. *Comparing Julia to performance portable parallel programming models for HPC // 2021 International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS)* (St. Louis, MO, USA, 15 November 2021).– IEEE.– 2021.– ISBN 978-1-6654-1119-6.– Pp. 94–105. doi ↑332

[152] *AMD EPYC™ 9004 Series Performance Package.*– 2022 (Accessed 9.10.2024).– 35 pp. URL ↑332, 345, 347

[153] *SPEC OMP 2012 Results* (Accessed 13.11.2024). URL ↑333

[154] *Tuning SPEComp2012 Performance on Lenovo ThinkSystem AMD Servers.*– 2023 (Accessed 20.10.2024).– 14 pp. URL ↑334

[155] *AMD Epyc 9004 "Genoa" buyers guide for CFD.*– 2022 (Accessed 21.10.2024). URL ↑334, 354, 355

[156] *All Published SPEC MPIM 2007 Results* (Accessed 6.05.2025). URL ↑335

[157] Larabel M. *Ampere Altra Max 128-Core CPU Is Priced Lower Than Flagship Xeon, EPYC™ CPUs.*– 2021 (Accessed 12.05.2025). URL ↑338

[158] Yalamanchi K., Vazhkudai S. *Boost HPC workloads with Micron DDR5 and 4th Gen AMD EPYC™ processors.*– 2022 (Accessed 9.02.2025). URL ↑339

[159] Rajasukumar A., Zhang T., Xu R., Chien A. A. *UpDown: a novel architecture for unlimited memory parallelism // MEMSYS '24: The International Symposium on Memory Systems* (Washington, DC, USA, 30 September 2024–3 October 2024), New York: ACM.– 2024.– ISBN 979-8-4007-1091-9.– Pp. 61–77. doi ↑342

[160] Nambiar R. *Performance Analysis of HPC Workloads: AMD EPYC™ with 3D V-Cache vs. Intel Xeon CPU Max with HBM.*– 2023 (Accessed 6.12.2024). URL ↑342, 365, 366, 457

[161] *Fujitsu Server PRIMERGY Memory performance of EPYC™ 9004 Series Processor (Genoa) based Systems White Paper v. 1.0.*– 2024 (Accessed 18.09.2024). URL ↑343

[162] *Workload-Based DDR5 Memory Guidance for Next-Generation PowerEdge Servers.*– 2023 (Accessed 19.12.2024).– 21 pp. URL ↑343, 344, 375

[163] Reguly I. Z. *Comparative evaluation of bandwidth-bound applications on the Intel Xeon CPU MAX Series // SC-W '23: Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0785-8.– Pp. 1236–1244. doi ↑344, 453, 457

[164] *Fujitsu Server PRIMERGY Performance Report. PRIMERGY RX2530 M6.*– 2023 (Accessed 11.02.2025). URL ↑345

[165] Sztemon M. *AMD Data Center Portfolio.*– 2024 (Accessed 9.10.2024).– 55 pp. URL ↑347, 357, 375

[166] *Fujitsu Server Performance Report PRIMERGY RX2530 M7 / RX2540 M7 v.1.5.*– 2024 (Accessed 31.12.2024). URL ↑347, 380, 414, 423, 428, 434

[167] Löff J., Griebler D., Mencagli G., Araujo G., Torquati M., Danelutto M., Fernandes L. G. *The NAS parallel benchmarks for evaluating C++ parallel programming frameworks on shared-memory architectures //* Future Generation Computer Systems.– 2021.– Vol. **125**.– No. C.– Pp. 743–757. doi ↑348

[168] Stegailov V., Dlinnova E., Ismagilov T., Khalilov M., Kondratyuk N., Makagon D., Semenov A., Simonov A., Smirnov G., Timofeev A. *Angara interconnect makes GPU-based Desmos supercomputer an efficient tool for molecular dynamics calculations //* The International Journal of High Performance Computing Applications.– 2019.– Vol. **33**.– No. 3.– Pp. 507–521. doi ↑349

[169] Ayala A., Tomov S., Haidar A., Dongarra J. heFFTe*: Highly efficient FFT for exascale // Computational Science — ICCS 2020: 20th International Conference.*– V. I (Amsterdam, The Netherlands, 3–5 June 2020), Berlin–Heidelberg: Springer-Verlag.– 2020.– ISBN 978-3-030-50370-3.– Pp. 262–275. doi ↑351

[170] Ayala A., Tomov S., Stoyanov M., Dongarra J. *Scalability issues in FFT computation* // *Parallel Computing Technologies: 16th International Conference*, PaCT 2021 (Kaliningrad, Russia, 13–18 September 2021), Lecture Notes in Computer Science.– vol. **12942**, Cham: Springer.– 2021.– ISBN 978-3-030-86358-6.– Pp. 279–287. ↑352

[171] Larabel M. *AMD EPYC™ 9684X Genoa-X Provides Incredible HPC Performance*.– 2023 (Accessed 25.11.2024).– 8 pp. ↑352, 355, 358, 370

[172] Moon G., Yi M., Jun E. *Design and operational concept of a cryogenic active intake device for atmosphere-breathing electric propulsion* // Aerospace Science and Technology.– 2024.– Vol. **151**.– id. 109300. ↑353

[173] German P., Marin O., Giudicelli G. L., Hansel J. E., Lindsay A. D., Yankura D. C., Charlot L., Freile R. O., Tano M. E. *Continued performance improvement and integration of MOOSE's thermal-hydraulics capabilities*, M3 Milestone Report, NoINL/RPT-24-80844-Rev000.– Idaho Falls, ID, USA: Idaho National Laboratory (INL).– 2024 (Accessed 29.11.2024).– 44 pp. ↑353

[174] Danciu B. A., Frouzakis C. E. *KinetiX: A performance portable code generator for chemical kinetics and transport properties*.– 2024.– 34 pp. arXiv 2411.02640 ↑353

[175] Gross S. *Simulation Hardware: The Sky is the Limit?*– 2022 (Accessed 26.11.2024). ↑353

[176] *OpenFOAM v12 User Guide*.– 2024 (Accessed 29.11.2024). ↑354

[177] Spisso I., Bna S., Amati G., Rossi G., Magugliani F. *HPC Benchmark Project (part I)*, OpenFoam Conference (Germany, 15–17 October 2019).– 2019 (Accessed 29.11.2024).– 27 pp. ↑354

[178] *OpenFOAM HPC Benchmark Suite*.– 2019 (Accessed 29.11.2024). ↑354

[179] *OpenFOAM and AMD 3d V-cache Technology Computational Fluid Dynamics*.– 2023 (Accessed 8.12.2024). ↑355

[180] Galeazzo F. C. C., Zhang F., Zirwes Th., Habisreuther P., Bockhorn H., Zarzalis N., Trimis D. *Implementation of an efficient synthetic inflow turbulence-generator in the open-source code OpenFOAM for 3D LES/DNS applications* // *High Performance Computing in Science and Engineering'20: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2020*, Cham: Springer.– 2021.– ISBN 978-3-030-80601-9.– Pp. 207–221. ↑355

[181] Galeazzo F. C. C., Weiß R. G., Lesnik S., Rusche H., Ruopp A. *Understanding Superlinear Speedup in Current HPC Architectures*.– 2024 (Accessed 30.11.2024).– 9 pp. ↑355

[182] Álvarez-Farré X., Gorobets A., Trias F. X. *A hierarchical parallel implementation for heterogeneous computing. Application to algebra-based CFD simulations on hybrid supercomputers* // Computers & Fluids.– 2021.– Vol. **214**.– id. 104768. ↑355

[183] Pareek S., Veena K., Naik M., Donthireddy P. *HPC Application Performance on Dell PowerEdge R6625 with AMD EPYC™-GENOA*.– 2023 (Accessed 30.11.2024). ↑355, 359, 360, 362, 364, 367

[184] *Supermicro, AMD, WEKA and NVIDIA deliver High Performance Ansys Turnkay Solution.*– 2024 (Accessed 5.10.2024).– 11 pp. URL ↑356

[185] Banchelli F., Garcia-Gasulla M., Houzeaux G., Mantovani F. *Benchmarking of state-of-the-art HPC Clusters with a Production CFD Code // PASC '20: Proceedings of the Platform for Advanced Scientific Computing Conference* (Geneva, Switzerland, 29 June 2020–1 July 2020), New York: ACM.– 2020.– ISBN 978-1-4503-7993-9.– id. 3.– 11 pp. doi ↑356

[186] Fernandez A. *Ansys Applications Technical Computing.*– 2022 (Accessed 8.05.2025). URL ↑356

[187] Fernandez A. et al. *Leadership Performance on ANSYS Applicattions Finite Element Analysis & Computational Fluid Dynamics.*– 2024 (Accessed 8.12.2024(**)). URL ↑356, 357

[188] Fernandez A., Manikonda A. *ANSYS LS-DYNA and AMD 3d V-cache Technology.*– 2023 (Accessed 11.10.2024). URL ↑357

[189] Fernandez A., Manikonda A. *ANSYS CFX and AMD 3D V-Cache Technology Computational Fluid Dynamics.*– 2023 (Accessed 8.12.2024). URL ↑357

[190] Fernandez A., Manikonda A. *ANSYS FLUENT and AMD 3D V-Cache Technology Computational Fluid Dynamics.*– 2023 (Accessed 11.10.2024). URL ↑357

[191] Wilfong B., Radhakrishnan A., Le Berre H. A., Abbott S., Budiardja R. D., Bryngelson S. H. *OpenACC offloading of the MFC compressible multiphase flow solver on AMD and NVIDIA GPUs.*– 2024.– 11 pp. arXiv 2409.10729 ↑358

[192] Fernandez A., Manikonda A. *WRF and AMD 3D V-cache technology weather forecasting.*– 2023 (Accessed 8.05.2025). URL ↑358, 359

[193] Larabel M. *AMD EPYC™ 9374F Linux Benchmarks — Genoa's 32-Core High Frequency CPU.*– 2022 (Accessed 28.09.2024).– 14 pp. URL ↑359

[194] Kashyap S., Kovouri S., Goyal P. *GROMACS on Amazon EC2 Hpc7a Instances.*– 2023 (Accessed 9.12.2024). URL ↑360

[195] Fernandez A., Manikonda A. *Molecular Dynamics on AMD EPYC™ 9754 Processors.*– 2023 (Accessed 9.12.2024). URL ↑360, 365, 438, 439

[196] Malik H., Nunes T. *4th Gen AMD EPYC™ Processors for Healthcare and Life Science.*– 2024 (Accessed 22.10.2024).– 13 pp. URL ↑365, 368, 375, 438

[197] Ramakrishna K., Lokamani M., Cangi A. *Electrical conductivity of warm dense hydrogen from Ohm's law and time-dependent density functional theory.*– 2024.– 9 pp. arXiv 2409.15160 ↑366

[198] Voorhis T., Vázquez-Mayagoitia Á., Verma P., Villa O., Vishnu A., Vogiatzis K. D., Wang D., Weare J. H., Williamson M. J., Windus T. L., Woliński K., Wong A. T., Wu Q., Yang C., Yu Q., Zacharias M., Zhang Z., Zhao Y., Harrison R. J. *NWChem: Past, present, and future* // The Journal of chemical physics.– 2020.– Vol. **152**.– No. 18.– id. 184102. doi ↑367

[199] Larabel M. *AMD EPYC™ 9374F Linux Benchmarks — Genoa's 32-Core High Frequency CPU.*– 2022 (Accessed 18.05.2025).– 14 pp. URL ↑367

[200] Larabel M. *AMD EPYC™ 9755 / 9575F / 9965 Benchmarks Show Dominating Performance.*– 2024 (Accessed 14.12.2024).– 14 pp. URL ↑367, 445, 469, 472, 473, 474, 475

[201] Larabel M. *AVX-512 Performance Comparison: AMD Genoa vs. Intel Sapphire Rapids & Ice Lake.*– 2023 (Accessed 15.12.2024).– 8 pp. URL ↑367

[202] Larabel M. *Intel Xeon Platinum 8592+ "Emerald Rapids" Linux Benchmarks.*– 2023 (Accessed 15.12.2024).– 10 pp. URL ↑367, 430, 431, 432, 433, 434, 435, 437, 438, 439, 440, 444, 447

[203] Hof J., Speed D. *LDAK-KVIK performs fast and powerful mixed-model association analysis of quantitative and binary phenotypes* // Nature Genetics.– 2025.– Vol. **57**.– Pp. 2116–2123. doi ↑368

[204] Bernardini G., van Iersel L., Julien E., Stougie L. *Inferring phylogenetic networks from multifurcating trees via cherry picking and machine learning* // Molecular Phylogenetics and Evolution.– 2024.– Vol. **199**.– id. 108137. doi ↑368

[205] Williams C. M., O'Connell J., Freyman W. A., 23andMe Research Team, Gignoux Ch. R., Ramachandran S., Williams A. L. *Phasing millions of samples achieves near perfect accuracy, enabling parent-of-origin classification of variants*, bioRxiv.– 2024. doi ↑368

[206] McKenna A., Hanna M., Banks E., Sivachenko A., Cibulskis K., Kernytsky A., Garimella K., Altshuler D., Gabriel S., Daly M., DePristo M. A. *The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data* // Genome research.– 2010.– Vol. **20**.– No. 9.– Pp. 1297–1303. doi ↑368

[207] Kendig K. I., Baheti S., Bockol M. A., Drucker T. M., Hart S. N., Heldenbrand J. R., Hernaez M., Hudson M. E., Kalmbach M. T., Klee E. W., Mattson N. R., Ross Ch. A., Taschuk M., Wieben E. D., Wiepert M., Wildman D. E., Mainzer L. S. *Sentieon DNASeq variant calling workflow demonstrates strong computational performance and accuracy* // Frontiers in genetics.– 2019.– Vol. **10**.– id. 736.– 7 pp. doi ↑368

[208] Malik H., Seniziaz M., Freed D., Gallagher B. *4th Gen AMD EPYC™ Priocessor Accelerate Genomics Data Processing with Sentieon.*– 2023 (Accessed 5.10.2024). URL ↑368

[209] *Nvidia Clara Parabrics Pipelines.*– 2021 (Accessed 15.12.2024). URL ↑368

[210] *AI Inferencing with AMD EPYC™ Processors*, White Paper.– 2024 (Accessed 09.01.2026).– 11 pp. URL ↑368

[211] Ha D. M. F., Alderliesten T., Bosman P. A. N. *Learning discretized Bayesian networks with GOMEA* // *Parallel Problem Solving from Nature — PPSN XVIII.*– V. III, PPSN 2024 (Hagenberg, Austria, 14–18 September 2024), Lecture Notes in Computer Science.– vol. **15150**, Cham: Springer.– 2024.– ISBN 978-3-031-70070-5.– Pp. 352–368. doi ↑368

[212] Nair K., Pandey A.-Ch., Karabannavar S., Arunachalam M., Kalamatianos J., Agrawal V., Gupta S., Sirasao A., Delaye E., Reinhardt S., Vivekanandham R., Wittig R., Kathail V., Gopalakrishnan P., Pareek S., Jain R., Kandemir M. T., Lin J.-L., Akbulut G. G., Das Ch. R. *Parallelization Strategies for DLRM Embedding Bag Operator on AMD CPUs* // IEEE Micro.– 2024.– Vol. **44**.– No. 6.– Pp. 44–51. ↑368

[213] Jocher G., Chaurasia A., Stoken A., Borovec J., NanoCode012, Kwon Y., Michael K., TaoXie, Fang J., Imyhxy, Lorna, Zeng Y., Wong C., Abhiram V., Montes D., Wang Zh., Fati C., Nadar J., Laughing, UnglvKitDe, Sonck V., Tkianai, YxNONG, Skalski P., Hogan A., Nair D., Strobel M., Jain M. *ultralytics/yolov5: v7.0 — YOLOv5 SOTA. Realtime instance segmentation.*– Zenodo.– 2022. ↑368

[214] Nambiar R. *4th Gen AMD EPYC™ Processors Deliver Exceptional Performance for AI Workloads.*– 2023 (Accessed 27.11.2024). ↑369

[215] Reddi V. J., Cheng Ch., Kanter D., Mattson P., Schmuelling G., Wu C.-J., Anderson B., Breughe M., Charlebois M., Chou W., Chukka R., Coleman C., Davis S., Deng P., Diamos G., Duke J., Fick D., Gardner J. S., Hubara I., Idgunji S., Jablin Th. B., Jiao J., John T. St., Kanwar P., Lee D., Liao J., Lokhmotov A., Massa F., Meng P., Micikevicius P., Osborne C., Pekhimenko G., Rajan A. T. R., Sequeira D., Sirasao A., Sun F., Tang H., Thomson M., Wei F., Wu E., Xu L., Yamada K., Yu B., Yuan G., Zhong A., Zhang P., Zhou Yu. *MLPerf inference benchmark* // *ISCA '20: Proceedings of the ACM/IEEE 47th Annual International Symposium on Computer Architecture* (Virtual Event, 30 May 2020–3 June 2020).– IEEE.– 2020.– ISBN 978-1-7281-4661-4.– Pp. 446–459. ↑369, 441, 442

[216] Gorbachev Y., Gorbachev Yu., Fedorov M., Slavutin I., Tugarev A., Fatekhov M. *OpenVINO deep learning workbench: Comprehensive analysis and tuning of neural networks inference* // *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (Seoul, Korea (South), 27–28 October 2019).– IEEE.– 2019.– ISBN 9798350307450.– Pp. 783–787 (Accessed 27.05.2025). ↑369

[217] Larabel M. *Intel Xeon Platinum 8490H "Sapphire Rapids" Performance Benchmarks.*– 2023 (Accessed 23.04.2025).– 14 pp. ↑370

[218] Rasley J., Rajbhandari S., Ruwase O., He Yu. *DeepSpeed: System optimizations enable training deep learning models with over 100 billion parameters* // *KDD '20: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Virtual Event, CA, USA, 6–10 July 2020), New York: ACM.– 2020.– ISBN 978-1-4503-7998-4.– Pp. 3505–3506. ↑370

[219] Kim S. H., Wang X., Fu Y. *Practical Strategies for low-cost LLM Deployments using 4th Gen AMD EPYC™ Processors.*– 2024 (Accessed 28.11.2024).– 9 pp. ↑370, 371

[220] *TPCx-AI Specification Version 2.0.0.*– 2024 (Accessed 27.11.2024).– 85 pp. ↑372

[221] Rajput A. *AMD EPYC™ 9004 Tunuing guide. Cloud Infrastructure and Datacenter Design & Configuration Rev. 1.03.*– 2024 (Accessed 18.12.2024).– 64 pp. URL ↑373, 466

[222] Gibby G. *Meet the New AMD EPYC™ 97X4 Processors, Optimized for Cloud-Native Workloads.*– 2023 (Accessed 18.12.2024). URL ↑373

[223] Nambiar R. *"Bergamo" 4th Gen AMD EPYC™ 97x4 Processors: Built for Cloud Native Workloads.*– 2023 (Accessed 18.12.2024). URL ↑373, 375

[224] *VMware Marketplace.*– 2024 (Accessed 15.12.2024). URL ↑373

[225] *VMmark 3.x Results.*– 2024 (Accessed 15.12.2024). URL ↑373

[226] *VMmark 4 Results.*– 2024 (Accessed 15.12.2024). URL ↑373

[227] *PostgreSQL pgbench.*– 2024 (Accessed 16.12.2024). URL ↑374

[228] *AMD EPYC™ Delivers Leadership MariaDB Perforamnce Relational Database.*– 2024 (Accessed 17.10.2024). URL ↑375

[229] Rajaram G., Porana J., Veerabhadrachary B., Kenchappanavara D. *Cloud-Native Workloads on AMD EPYC™ 9754 Processors.*– 2023 (Accessed 09.01.2026).– 13 pp. URL ↑375

[230] Nambiar R. *4th Gen AMD EPYC™ 8004 Series Processors Designed for Intelligent Edge.*– 2023 (Accessed 18.12.2024). URL ↑376

[231] *Intel Xeon Platinum 8380 Processor* (Accessed 5.08.2024(*)). URL ↑377

[232] Schuh H. N., Krishnamurthy A., Culler D., Levy H. M., Rizzo L., Khan S., Stephens B. E. *CC-NIC: a cache-coherent interface to the NIC // ASPLOS '24: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems.*– V. 1, New York: ACM.– 2024.– ISBN 979-8-4007-0372-0.– Pp. 52–68. doi ↑378

[233] Mulnix D. L. *Intel Xeon Processor Scalable Family Technical Overview.*– 2022 (Accessed 5.08.2024(*)). URL ↑379, 381, 382, 383

[234] *Product Naming Convention for the 4th and 5th Gen Intel Xeon Scalable Processors.*– 2024 (Accessed 10.08.2024(*)). URL ↑380

[235] *Intel Xeon Scalable Processors Numbers and Suffixes.*– 2024 (Accessed 13.01.2025(*)). URL ↑379, 380

[236] Watts D. *Lenovo ThinkSystem SR950 V3 Server Product Guide.*– 2025 (Accessed 09.01.2026).– 80 pp. URL ↑381

[237] *Intel Scalable I/O Virtualization.*– 2020 (Accessed 09.01.2026).– 29 pp. URL ↑383

[238] Nassif N., Munch A. O., Molnar C. L., Pasdast G., Lyer S. V., Yang Z. *Sapphire Rapids: The next-generation Intel Xeon Scalable processor // 2022 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 20–26 February 2022).– IEEE.– 2022.– ISBN 978-1-6654-2801-9.– Pp. 44–46. doi ↑383, 387, 391, 392

[239] Biswas A. *Sapphire Rapids.*– HotChips.– 2021 (Accessed 12.05.2025).– 22 pp. URLIntel Arijit.pdf ↑383, 384, 392

[240] *Network-Optimized 4th Gen Intel Xeon Scalable Processors Product Brief* (Accessed 5.08.2024(*)). URL ↑383

[241] *Intel® 64 and IA-32 Architectures Optimization Reference Manual Documentation Changes*, Aug 2023.– 2023 (Accessed 5.08.2024).– 379 pp. URL ↑384, 385, 386, 387

[242] Mujtaba H. *AMD Zen 4 For Ryzen 7000 & Intel Raptor Cove For Raptor Lake CPUs Have Almost Similar IPC.*– 2022 (Accessed 15.02.2025). URL ↑386

[243] Mulnix D. et al. *Technical Overview Of The 4th Gen Intel Xeon Scalable processor family.*– 2022 (Accessed 5.08.2024(*)). URL ↑387, 388, 390, 391, 392, 393, 394, 395, 396, 398

[244] *Intel 64 and IA-32 Architectures Software Developer Manuals Upd.* (Accessed 5.08.2024(*)). URL ↑387

[245] *Intel Architecture Instruction Set Extensions and Future Features Programming Reference.*– 2024 (Accessed 6.08.2024(*)). URL ↑388, 389, 390

[246] *Intel 64 and IA-32 Architectures Optimization Reference Manual.*– V. 1.– 2023 (Accessed 09.01.2026).– 963 pp. URL ↑389, 390

[247] Dizani F., Ghanbari A., Kalyanapu J., Asher D., Ajorpaz S. M. *Thor: A non-speculative value dependent timing side channel attack exploiting Intel AMX* // IEEE Computer Architecture Letters.– 2025.– Vol. **24**.– No. 1.– Pp. 69–72. doi ↑389

[248] Cutress I. *Intel Xeon Sapphire Rapids: How To Go Monolithic with Tiles.*– 2021 (Accessed 23.04.2025). URL ↑391

[249] Mahajan R., Sankman R., Patel N., Kim D.-W., Aygun K., Qian Zh. *Embedded multi-die interconnect bridge (EMIB) — a high density, high bandwidth packaging interconnect* // *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)* (Las Vegas, NV, USA, 31 May 2016–03 June 2016).– IEEE.– 2016.– ISBN 9781509012053.– Pp. 557–565. doi ↑392

[250] Mahajan R., Sankman R., Aygun K., Qian Zh., Dhall A., Rosch J., Mallik D., Salama I. *Embedded Multi-die Interconnect Bridge (EMIB). A localized, high density, high bandwidth packaging interconnect* // *Advances in Embedded and Fan-Out Wafer-Level Packaging Technologies*, Ch. 23, eds. Keser B., Kroehnert S..– Wiley.– 2019.– ISBN 9781119314134.– Pp. 487–499. doi ↑392

[251] Duan G., Kanaoka Y., McRee R., Nie B., Manepalli R. *Die embedding challenges for EMIB advanced packaging technology* // *2021 IEEE 71st Electronic Components and Technology Conference (ECTC)* (San Diego, CA, USA, 01 June 2021–04 July 2021).– IEEE.– 2021.– ISBN 9781665431200.– Pp. 1–7. doi ↑392

[252] Lee C. C., Chang Y. W. *Floorplanning for embedded multi-die interconnect bridge packages* // *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)* (San Francisco, CA, USA, 28 October 2023–02 November 2023).– IEEE.– 2023.– ISBN 9798350322262.– Pp. 1–8. doi ↑392

[253] Mujtaba H. *Intel Sapphire Rapids-SP Xeon Server CPU Detailed: Quad-Tile Chiplet Design With EMIB, 56 Cores, 112 Threads, CXL 1.1, DDR5, HBM & PCIe 5.0 Support.*– 2021 (Accessed 10.05.2025). URL ↑392

[254] Kuper R., Jeong I., Yuan Y., Wang R., Ranganathan N., Rao N., Hu J., Kumar S., Lantz P., Kim N. S. *A quantitative analysis and guidelines of data streaming accelerator in modern Intel Xeon Scalable Processors // ASPLOS '24: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems.*– V. 2 (La Jolla, CA, USA, 27 April 2024–1 May 2024), New York: ACM.– 2024.– ISBN 979-8-4007-0385-0.– Pp. 37–54. ↑395

[255] Berthold A., Fürst C., Obersteiner A., Schmidt L., Habich D., Lehner W., Schirmeier H. *Demystifying Intel Data Streaming Accelerator for in-memory data processing // DIMES '24: Proceedings of the 2nd Workshop on Disruptive Memory Systems* (Austin, TX, USA, 3 November 2024), New York: ACM.– 2024.– ISBN 979-8-4007-1303-3.– Pp. 9–16. ↑395

[256] *Proof Points of Intel® Dynamic Load Balancer (Intel® DLB)* (Accessed 10.08.2024(*)). ↑395, 396

[257] Sexton R., Coyle D., Przychodni D. *Guidelines for Optimizing Power Consumption of vCMTS Deployments on Intel Xeon Scalable Processors* (Accessed 19.08.2024).– 15 pp. ↑396, 397

[258] Pattan R., Khan H. *Intel Turbo Boost Technology Configure Per Core Turbo Overview.*– 2022 (Accessed 24.04.2025).– 6 pp. ↑397

[259] Huffstetler J. *Sustainability In Focus with 4th Gen Intel Xeon Processors.*– 2023 (Accessed 19.03.2025). ↑397

[260] Rieger E. *Half-Socket VM support for SAP HANA on vSphere 8 and 4th Gen Intel® Xeon® Scalable Processors (Sapphire Rapids).*– 2024 (Accessed 13.08.2024). ↑397

[261] *Intel ® Virtualization Technology for Directed I/O Architecture Specification*, Rev. 4.1.– 2023 (Accessed 13.08.2024(*)). ↑398

[262] Moghimi D. *Downfall: Exploiting speculative data gathering // SEC '23: Proceedings of the 32nd USENIX Conference on Security Symposium* (Anaheim, CA, USA, 9–11 August 2023), Berkeley: USENIX Ass..– 2023.– ISBN 978-1-939133-37-3.– Pp. 7179–7193.– id. 402. ↑399

[263] *Intel® Trust Domain Extensions*, White Paper.– 2022 (Accessed 14.08.2024).– 9 pp. ↑399, 400, 401, 402

[264] Sunny A., Shrivastava N., Sarangi S. R. *SecScale: a scalable and secure trusted execution environment for servers.*– 2024.– 13 pp. arXiv 2407.13572 ↑399, 400, 402

[265] Akram A., Giannakou A., Akella V., Lowe-Power J., Peisert S. *Performance analysis of scientific computing workloads on general purpose TEEs // 2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (Portland, OR, USA, 17–21 May 2021).– IEEE.– 2021.– ISBN 978-1-6654-1156-1.– Pp. 1066–1076. ↑399, 400

[266] Miladinović D., Milaković A., Vukasović M.„ Stanisavljević Ž., Vuletić P. *Secure Multiparty Computation Using Secure Virtual Machines* // Electronics.– 2024.– Vol. **13**.– No. 5.– id. 991.– 25 pp. ↑400

[267] Demigha O., Larguet R. *Hardware-based solutions for trusted cloud computing* // Computers & Security.– 2021.– Vol. **103**.– id. 102117. ↑400, 402

[268] Li M., Wilke L., Wichelmann J., Eisenbarth Th., Teodorescu R., Zhang Y. *A systematic look at ciphertext side channels on AMD SEV-SNP* // *2022 IEEE Symposium on Security and Privacy (SP)* (San Francisco, CA, USA, 22–26 May 2022).– IEEE.– 2022.– ISBN 9781665413176.– Pp. 337–351. ↑400

[269] *Intel® Trust Domain Extension Research and Assurance*, Version 2.0.– 2024 (Accessed 15.08.2024(*)). ↑400, 401

[270] Wang W., Song L., Mei B., Liu Sh., Zhao Sh., Yan Sh., Wang X.-F., Meng D., Hou R. *NestedSGX: bootstrapping trust to enclaves within confidential VMs.*– 2024.– 16 pp. arXiv 2402.11438 ↑402

[271] Stephan C., Rahman R. *Balanced Memory Configurations for 2-Socket Servers with 4th and 5th Gen Intel Xeon Scalable Processors.*– 2024 (Accessed 21.08.2024).– 13 pp. ↑402

[272] Lesmanne S. et al. *Rise of Volumetric Data and Scale-Up Enterprise Computing.*– 2021 (Accessed 8.08.2024(**)). ↑402

[273] *HPE Compute Scale-up Server 3200.*– 2023 (Accessed 10.08.2024).– 7 pp. ↑403

[274] *HPE Compute Scale-up 3200 Server Performance Tuning Guide* (Accessed 12.05.2025). ↑403

[275] *Microsoft Eagle ND v5 System* (Accessed 21.08.2024). ↑403

[276] Turisini M., Amati G., Cestari M. *LEONARDO: A pan-European pre-exascale supercomputer for HPC and AI applications.*– 2023.– 16 pp. arXiv 2307.16885 ↑403

[277] *MareNostrum 5 System overview* (Accessed 21.08.2024). ↑403, 404

[278] Shipman G. M., Swaminarayan S., Grider G., Lujan J., Zerr R. J. *Early Performance Results on 4th Gen Intel® Xeon® Scalable Processors with DDR and Intel® Xeon® processors, codenamed Sapphire Rapids with HBM.*– 2022.– 5 pp. arXiv 2211.05712 ↑404, 409, 453

[279] *Technical Overview Of The Intel Xeon Scalable processor Max Series.*– 2022 (Accessed 11.08.2024(*)). ↑405, 406

[280] Kuo S., Cheng J. *Implementing High Bandwidth Memory and Intel Xeon Processors Max Series on Lenovo ThinkSystem Servers.*– 2024 (Accessed 24.09.2024).– 21 pp. ↑404, 406, 407, 408, 454, 455

[281] McCalpin J. D. *Bandwidth limits in the Intel Xeon Max (Sapphire Rapids with HBM) processors* // *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, 21–25 May 2023), Lecture Notes in Computer Science.– vol. **13999**, Cham: Springer.– 2023.– ISBN 978-3-031-40842-7.– Pp. 403–413. ↑405, 455, 456, 457, 465

[282] *Intel® Xeon® CPU Max Series Configuration and Tuning Guide.*– 2023 (Accessed 12.08.2024).– 35 pp. ↑406

[283] Fukazawa K., Takahashi R. *Performance evaluation of the fourth-generation Xeon with different memory characteristics* // *HPCAsia '24 Workshops: Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region Workshops* (Nagoya, Japan, 25–27 January 2024), New York: ACM.– 2024.– ISBN 979-8-4007-1652-2.– Pp. 55–62. ↑407, 425, 453, 455, 457, 461, 462

[284] Sanca V., Ailamaki A. *Post-Moore's Law Fusion: high-bandwidth memory, accelerators, and native half-precision processing for CPU-local analytics*, Joint Workshop sat 49th International Conference on Very Large Data Bases (VLDBW'23) — Workshop on Accelerating Analytics and Data Management Systems (ADMS'23) (Vancouver, Canada, August 28–September 1 2023), CEUR Workshop Proceedings.– vol. **3462**.– 2023.– id. ADMS1 (Accessed 13.08.2024).– 13 pp. URL ↑408, 453, 454

[285] Munch A. O., Nassif N., Molnar C. L., Crop J., Gammack R., Joshi Ch. P. *2.3 Emerald Rapids: 5th-Generation Intel® Xeon® Scalable Processors // 2024 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 18–22 February 2024).– IEEE.– 2024.– ISBN 979-8-3503-0621-7.– Pp. 40–42. doi ↑409, 411, 412

[286] *5th Gen Intel® Xeon® Processors Product Brief.*– 2023 (Accessed 18.08.2024(*)). URL ↑409, 411, 412, 413

[287] *5th Gen Intel Xeon Scalable Processor XCC (Codename Emerald Rapids) Uncore Performance Monitoring Guide*, Rev. 001.– 2024 (Accessed 29.10.2024).– 353 pp. URL ↑409

[288] Bai C., Huang J., Wei X., Ma Yu., Li S., Zheng H., Yu B., Xie Yu. *ArchExplorer: Microarchitecture exploration via bottleneck analysis // MICRO '23: Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture* (Toronto, ON, Canada, 28 October 2023–1 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0329-4.– Pp. 268–282. doi ↑411

[289] Alcorn P. *Intel 'Emerald Rapids' 5th-Gen Xeon Platinum 8592+ Review: 64 Cores, Tripled L3 Cache and Faster Memory Deliver Impressive AI Performance.*– 2023 (Accessed 7.01.2025). URL ↑411, 412, 413, 430, 436, 445

[290] *5th Gen Intel® Xeon® Scalable Processors* (Accessed 22.08.2024(*)). URL ↑411

[291] Morgan T. P. *Intel "Emerald Rapids" Xeon SPs: A Little More Bang, A Little Less Bucks.*– 2023 (Accessed 20.08.2024). URL ↑411, 413

[292] Mujtaba H. *Intel 5th Gen Xeon CPUs Official: Emerald Rapids Compatible With Sapphire Rapids, Up To 64 Cores, 320 MB Cache, Prices Detailed.*– 2023 (Accessed 17.08.2024). URL ↑412

[293] *Lenovo Launches New and Updated Servers with 5th Gen Intel Xeon Scalable Processors.*– 2023 (Accessed 21.02.2025).– 10 pp. URL ↑412

[294] *Performance Tuning Guide.*– 2024 (Accessed 2.01.2024). URL ↑414

[295] *CPU Practice Guide.*– 2024 (Accessed 2.01.2024). URL ↑414

[296] *4th Generation Intel Xeon Scalable Processors* (Accessed 4.02.2025(*)). URL ↑414

[297] *5th Generation Intel® Xeon® Scalable Processors* (Accessed 4.02.2025(*)). URL ↑414

[298] *Accelerate with Xeon.*– 2023 (Accessed 22.12.2024).– 69 pp. URL ↑414, 415

[299] Afzal A., Hager G., Wellein G. *SPEChpc 2021 benchmarks on Ice Lake and Sapphire Rapids Infiniband clusters: A performance and energy case study // SC-W '23: Proceedings of the SC '23 Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0785-8.– Pp. 1245–1254. ↑419, 420

[300] Laros III J. H., Pedretti K., Kelly S. M., Shu W., Ferreira K., Vandyke J., Vaughan C. *Energy delay product // Energy-Efficient High Performance Computing: Measurement and Tuning*, SpringerBriefs in Computer Science, London: Springer.– 2013.– ISBN 978-1-4471-4491-5.– Pp. 51–55. ↑419

[301] Laukemann J., Gruber T., Hager G., Oryspayev D., Wellein G. *CloverLeaf on Intel multi-core CPUs: A case study in write-allocate evasion // 2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (San Francisco, CA, USA, 27–31 May 2024).– IEEE.– 2024.– ISBN 9798350387124.– Pp. 350–360. ↑421

[302] *Fujitsu server performance report PRIMERGY TX2550 M7.*– 2024 (Accessed 24.02.2025). ↑423

[303] *Fujitsu performance report PRIMERGY CX2550 M7 / CX2560 M7 v.1.3.*– 2024 (Accessed 22.02.2025). ↑423, 424, 428, 434, 461

[304] *Memory performance of Xeon Scalable Processor (Sapphire Rapids / Emerald Rapids) based Systems v.1.1.*– 2024 (Accessed 25.09.2024). ↑424

[305] Gray B., Dumas D., Shakshober D., Mehta M. *Red Hat Enterprise Linux performance results on 5th Gen Intel® Xeon® Scalable Processors.*– 2024 (Accessed 22.12.2024). ↑425, 434

[306] *Fujitsu server performance report PRIMEQUEST 4400E v. 1.0.*– 2023 (Accessed 22.12.2024). ↑426, 429

[307] Tatebe O. *Data and computational science driven by Persistent Memory supercomputer Pegasus.*– 2023 (Accessed 5.03.2025).– 19 pp. ↑426

[308] Croci M., Wells G. N. *Mixed-precision finite element kernels and assembly: Rounding error analysis and hardware acceleration.*– 2024.– 37 pp. arXiv:2410.12614 ↑426

[309] *Using Intel's new built-in AI acceleration engines.*– 2024 (Accessed 31.07.2024(*)).– 57 pp. ↑426, 427

[310] Kim H., Ye G., Wang N., Yazdanbakhsh A., Kim N. S. *Exploiting Intel® advanced matrix extensions (AMX) for large language model inference* // IEEE Computer Architecture Letters.– 2024.– Vol. **23**.– No. 1.– Pp. 117–120. ↑426, 441

[311] Georganas E., Kalamkar D., Voronin K., Kundu A., Noack A., Pabst H. *Harnessing deep learning and HPC kernels via high-level loop and tensor abstractions on CPU architectures // 2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (San Francisco, CA, USA, 27–31 May 2024).– IEEE.– 2024.– ISBN 9798350387124.– Pp. 950–963. ↑426, 447, 448

[312] *Отчет о тестировании масштабируемых процессоров Intel X Xeon 4-го поколения*, https://axel.as-1.co.jp/images/pdf/hpcs/benchmark03.pdf.– 2023 (Accessed 8.01.2025).– 20 с. (японский) ↑429

[313] Marjanović V., Gracia J., Glass C. W. *Performance modeling of the HPCG benchmark* // *High Performance Computing Systems. Performance Modeling, Benchmarking, and Simulation*, 5th International Workshop, PMBS 2014 (New Orleans, LA, USA, 16 November 2014), Lecture Notes in Computer Science.– vol. **8966**, Cham: Springer.– 2015.– ISBN 978-3-319-17247-7.– Pp. 172–192. doi ↑430

[314] Pareek S., Naik M., Veena K. *16G PowerEdge platform BIOS characterization for HPC with Intel Sapphire Rapids.*– 2023 (Accessed 8.01.2025). URL ↑430, 431

[315] Heroux M., Dongarra J., Luszczek P. *HPCG technical specification*, Sandia Report SAND2013-8752.– Sandia National Laboratories.– 2013 (Accessed 8.03.2025).– 21 pp. URL ↑431

[316] Fogli A., Zhao B., Pietzuch P., Bandle M., Giceva J. *OLAP on modern chiplet-based Processors* // Proceedings of the VLDB Endowment.– 2024.– Vol. **17**.– No. 11.– Pp. 3428–3441. doi ↑433

[317] Plana-Riu J., Xavier Trias F., Alsalti-Baldellou A., Colomer G., Oliva A. *Performance analysis of parallel-in-time techniques in modern supercomputers* // *ECCOMAS: 9th European Congress on Computational Methods in Applied Sciences and Engineering* (Lisbon, Portugal, 3–7 June 2024).– International Centre for Numerical Methods in Engineering (CIMNE).– 2024.– ISBN 978-84-127483-6-9 (Accessed 20.03.2025).– 10 pp. URL ↑434

[318] Larabel M. *Ubuntu 24.04 + Linux 6.9 Intel & AMD server performance.*– 2024 (Accessed 18.03.2025).– 10 pp. URL ↑435, 436, 438, 466

[319] Larabel M. *Intel 5th Gen Xeon performance benchmarks: impressive efficiency gains with "Optimized Power Mode".*– 2023 (Accessed 7.02.2025).– 10 pp. URL ↑435

[320] Pareek S., Veena K., Naik M., Donthireddy P. *HPC Application Performance on Dell PowerEdge C6620 with Intel 8480+ SPR.*– 2023 (Accessed 30.11.2024). URL ↑436, 439

[321] *NAMD GPU benchmarks and hardware recommendations.*– 2024 (Accessed 24.03.2025). URL ↑438

[322] Prabhakaran A. et al. *Scalable End-to-End Enterprise AI on 4th Gen Intel Xeon Scalable Processors*, White Paper.– 2023 (Accessed 7.02.2025(*)). URL ↑440

[323] Huang W., Wang D., Zhou S., Li C., Xie Y., He P. *Accelerating deep learning workloads with advanced matrix extensions* // *2023 5th International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)* (Hangzhou, China, 15–17 December 2023).– IEEE.– 2023.– ISBN 9798350359923.– Pp. 70–73. doi ↑441

[324] de Oliveira Silva J. V., Tahmid T., da Silva Pereira W. *Low precision and efficient programming languages for sustainable AI: Final report for the summer project of 2024*, NoNREL/TP-2C00-90910.– Golden, CO, USA: National Renewable Energy Laboratory (NREL).– 2024 (Accessed 10.05.2025).– 27 pp. URL ↑441

[325] AbouElhamayed A. F., Dotzel J., Akhauri Y., Chang C.-C., Gobriel S., Pablo Muñoz J., Chua V. S., Jain N., Abdelfattah M. S. *SparAMX: Accelerating compressed LLMs token generation on AMX-powered CPUs*.– 2025.– 14 pp. arXiv 2502.12444 ↑441

[326] Quinn D., Nouri M., Patel N., Salihu J., Salemi A., Lee S., Zamani H., Alian M. *Accelerating retrieval-augmented generation // ASPLOS '25: Proceedings of the 30th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*.– V. 1 (Rotterdam, Netherlands, 30 March 2025–3 April 2025), New York: ACM.– 2025.– ISBN 979-8-4007-0698-1.– Pp. 15–32. ↑441

[327] Saha S. *Taking on AI inferencing with 5th Gen Intel Xeon Scalable Processors*.– 2024 (Accessed 2.01.2025). ↑441

[328] *OpenVINO 2024.6* (Accessed 17.01.2025). ↑441

[329] *Benchmark suite results. MLPerf inference: datacenter*.– 2024 (Accessed 14.01.2025). ↑441, 442

[330] Eassa A., Nanjappa A., Jiang Zh., Zhang Y., Yang J., Kong Z., Xu Sh. *NVIDIA Blackwell Platform Sets New LLM Inference Records in MLPerf Inference v4.1*.– 2024 (Accessed 14.01.2025). ↑442

[331] Zhang T., Patel B., Sundararajan N., Engh J., Qiu Y., Tsai L., Iyengar T., Chukka R. *MLPerf™ Inference 4.0 on Dell PowerEdge Server with Intel® 5th Generation Xeon® CPU*.– 2024 (Accessed 25.03.2025). ↑442

[332] Zhang T., Patel B. *Running LLMs on Dell PowerEdge Servers with Intel® 4th Generation Xeon® CPUs*.– 2024 (Accessed 4.12.2025). ↑442

[333] Jithin V. G., Ditto P. S., Adarsh M. S. *Inference acceleration for large language models on CPUs*.– 2024.– 9 pp. arXiv 2406.07553 ↑442

[334] Chu A., Vance A. *Achieve Leading AI Performance with 5th Gen Intel Xeon Processors and Open Source AI Software*, upd. 6/14/2024 (Accessed 2.01.2024(*)). ↑443

[335] Santana Pacheco F. *Performance evaluation of object detection in different architectures*, Master in High Performance Computing.– SISSA.– 2023 (Accessed 12.01.2025).– 62 pp. ↑443, 453

[336] Terven J., Cordova-Esparza D. *A comprehensive review of YOLO: From YOLOv1 and beyond*.– 2023.– 36 pp. arXiv 2304.00501 ↑443

[337] *Performance Data for Intel AI Data Center Products* (Accessed 15.01.2025(*)). ↑445

[338] *Performance Data for Intel AI Data Center Products* (Accessed 15.01.2025(*)). ↑445

[339] Abuzaina M., Bhuiyan A., Minakshi M. *Improve performance of TensorFlow AI Workloads Using 5th Gen Intel Xeon Scalable Processors*.– 2023 (Accessed 15.01.2025(*)). ↑445, 446

[340] Jiang X., Minakshi M., Poornachandran R., Najnin Sh. *Power efficient deep learning acceleration using Intel Xeon Processors // 2024 IEEE High Performance Extreme Computing Conference (HPEC)* (Wakefield, MA, USA, 23–27 September 2024).– IEEE.– 2024.– Pp. 1–6. doi ↑446

[341] Hariharan R. *Leadership XGBoost Performance Outperforminh 5th Gen Intel Xeon.*– 2024 (Accessed 08.01.2026). URL ↑448

[342] Gamblin T., LeGendre M., Collette M. R., Lee G. L., Moody A., de Supinski B. R. *The Spack package manager: bringing order to HPC software chaos // SC '15: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Austin, TX, USA, 15–20 November 2015).– IEEE.– 2015.– ISBN 978-1-5090-0273-3.– Pp. 1–12. doi ↑449

[343] *NVIDIA HPC application performance.*– 2024 (Accessed 9.10.2024). URL ↑452

[344] *NAMD GPU benchmarks and hardware recommendations.*– 2024 (Accessed 12.01.2025). URL ↑452

[345] Linford J. *Developing of Nvidia superchips.*– 2024 (Accessed 8.01.2025).– 73 pp. URL ↑452

[346] Olenik G., Koch M., Boutanios Z., Anzt H. *Towards a platform-portable linear algebra backend for OpenFOAM* // Meccanica.– 2024.– Vol. **60**.– Pp. 1659–1672. doi ↑452

[347] Halbiniak K., Rojek K., Iserte S., Wyrzykowski R. *Unleashing the potential of mixed precision in AI-accelerated CFD simulation on Intel CPU/GPU architectures // 24th International Conference on Computational Science* (Málaga, Spain, 2–4 July 2024), Lecture Notes in Computer Science.– vol. **14837**, Cham: Springer.– ISBN 978-3-031-63777-3.– Pp. 203–217. doi ↑453

[348] Sfiligoi I., Belli E. A., Candy J. *The benefits of HBM memory for CPU-based fusion simulations // PEARC '24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), New York: ACM.– 2024.– ISBN 979-8-4007-0419-2.– id. 103.– 2 pp. doi ↑453, 462

[349] Martin J. E., Feldman C., Calder A., Curtis T., Siegmann E., Carlson D., Gonzalez R., Wood D., Harrison R., Coskun F. *Benchmarking with Supernovae: A performance study of the FLASH code // PEARC '24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), New York: ACM.– 2024.– ISBN 979-8-4007-0419-2.– id. 8.– 9 pp. doi ↑453

[350] Arslan V. *HBM Contribution to Intel Sapphire Rapids for Geophysical Workloads // EAGE Seventh High Performance Computing Workshop*, Volume 2023 (Lugano, Switzerland, 25–27 September 2023).– European Association of Geoscientists & Engineers.– 2023.– Pp. 1–5. doi ↑453

[351] Li Z., Rickert G., Zheng N., Zhang Zh., Özgen-Xian I., Caviedes-Voullième D. *SERGHEI v2.0: introducing a performance-portable, high-performance three-dimensional variably-saturated subsurface flow solver (SERGHEI-RE)* // Geoscientific Model Development.– 2025.– Vol. **18**.– No. 2.– Pp. 547–562. doi ↑453

[352] Piroozan N., Kumar N. *Enabling performant thermal conductivity modeling with DeePMD and LAMMPS on CPUs* // *Proceedings of the SC'23 Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis* (Denver, CO, USA, 12–17 November 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0785-8.– Pp. 88–94. ↑453

[353] Ibeid H. et al. *Performance Analysis of HPC applications on the Aurora Supercomputer: Exploring the Impact of HBM-Enabled Intel Xeon Max CPUs.*– 2025.– 11 pp. arXiv 2504.03632 ↑453, 454, 481

[354] McCalpin J. D. *Bandwidth limits in the Intel Xeon Max (Sapphire Rapids with HBM) processors* // *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, May 21–25 2023), Lecture Notes in Computer Science.– vol. **13999**, Cham: Springer.– 2023.– ISBN 978-3-031-40842-7.– Pp. 403–413 (Accessed 26.02.2025). ↑455, 456, 457, 465, 470

[355] Wang Y., McCalpin J. D., Li J., Cawood M., Cazes J., Chen H., Koesterke L., Liu H., Lu Ch.-Y., McLay R., Milfield K., Ruhela A., Semeraro D., Zhang W. *Application performance analysis: a report on the impact of memory bandwidth* // *High Performance Computing*, ISC High Performance 2023 (Hamburg, Germany, May 21–25 2023), Lecture Notes in Computer Science.– vol. **13999**, Cham: Springer.– 2023.– ISBN 978-3-031-40842-7.– Pp. 339–352 (Accessed 18.05.2025). ↑457, 462, 465

[356] Ristow Hadlich R., Verma G., Curtis T., Siegmann E., Assanis D. *A64FX enables engine decarbonization using deep learning* // *PEARC'24: Practice and Experience in Advanced Research Computing 2024: Human Powered Computing* (Providence, RI, USA, 21–25 July 2024), New York: ACM.– 2024.– ISBN 979-8-4007-0419-2.– id. 52.– 5 pp. ↑458

[357] Simakov N. A., Jones M. D., Furlani T. R., Siegmann E., Harrison R. J. *First impressions of the NVIDIA Grace CPU Superchip and NVIDIA Grace Hopper Superchip for scientific workloads* // *Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region Workshops* (Nagoya, Japan, 25–27 January 2024), New York: ACM.– 2024.– ISBN 979-8-4007-1652-2.– Pp. 36–44. ↑459, 460, 481

[358] Larabel M. *Intel Xeon Max Enjoying some performance gains with Linux 6.6.*– 2023 (Accessed 11.04.2025). ↑462

[359] Larabel M. *Intel Xeon Max 9480/9468 show significant uplift in HPC & AI workloads with HBM2e.*– 2023 (Accessed 9.04.2025). ↑462, 463, 464, 465

[360] Kutzner C., Miletić V., Rodríguez K. P., Rampp M., Hummer G., de Groot B. L., Grubmüller H. *Scaling of the GROMACS molecular dynamics code to 65k CPU cores on an HPC cluster* // Journal of Computational Chemistry.– 2025.– Vol. **46**.– No. 5.– id. e70059. ↑464

[361] *Mdsv3 Very High Memory series.*– 2024 (Accessed 13.04.2025). ↑466

[362] *Серия размеров Dlsv5.*– 2024 (Accessed 13.04.2025). ↑466

[363] Britton L., Malamed S. *Public preview: new AMD-based VMs with increased performance, Azure Boost, and NVMe support.*– 2023 (Accessed 6.10.2024). URL ↑466

[364] Yun C. *New – Amazon EC2 Hpc7a instances powered by 4th Gen AMD EPYC™ processors optimized for high performance computing.*– 2023 (Accessed 13.04.2025). URL ↑467

[365] *Developer clouds for accelerated computing* (Accessed 13.04.2025(*)). URL ↑467

[366] Thiagarajan M. *Announcing World's Largest, first Zettascale AI supercomputer in the Cloud.*– 2024 (Accessed 13.04.2025). URL ↑467

[367] Morgan T. P. *Intel rounds out "Granite Rapids" Xeon 6 with a slew of chips.*– 2025 (Accessed 17.05.2025). URL ↑467, 474

[368] Powell M. D., Fleming P., Krishna V. I., Lakkakula N., Ravisundar S., Mosur P. *Intel Xeon 6 product family* // IEEE Micro.– 2025.– Vol. **45**.– No. 3.– Pp. 31–40. doi ↑467, 468

[369] Gianos C. *Architecting for flexibility and value with Next Gen Intel® Xeon® processors* // *2023 IEEE Hot Chips 35 Symposium (HCS)* (Palo Alto, CA, USA, 27–29 August 2023).– IEEE.– 2023.– ISBN 9798350339086.– Pp. 1–15. doi ↑468

[370] *5th Gen AMD EPYC™ Processor Architecture 1st ed*, whitepaper.– 2025 (Accessed 25.04.2025).– 19 pp. URL ↑469

[371] *AMD EPYC™ 9005 Processor Architecture Overview*, whitepaper.– 2025 (Accessed 25.04.2026).– 11 pp. URL ↑469

[372] Singh T., Oliver S., Rangarajan S., Southard S., Henrion C., Schaefer A. *"Zen 5": The AMD high-performance 4nm x86-64 microprocessor core* // *2025 IEEE International Solid-State Circuits Conference (ISSCC)* (San Francisco, CA, USA, 16–20 February 2025).– IEEE.– 2025.– Pp. 1–3. doi ↑469

[373] Cohen B., Subramony M., Clark M. *Next generation "Zen 5" core* // *2024 IEEE Hot Chips 36 Symposium (HCS)* (Stanford, CA, USA, 25–27 August 2024).– IEEE.– 2024.– Pp. 1–27. doi ↑469, 470

[374] *Zen 5 – Microarchitectures – AMD* (Accessed 26.04.2025). URL ↑469

[375] Larabel M. *Intel Xeon 6980P 1S performance with DDR5-6400/MRDIMM-8800* (Accessed 24.11.2024). URL ↑469

[376] *Intel® Xeon® 6*, Performance Index.– 2025 (Accessed 26.04.2025(*)). URL ↑470, 471

[377] Sehgal R. et al. *Optimizing system memory bandwidth with Micron CXL memory expansion modules on Intel Xeon 6 processors.*– 2024.– 7 pp. arXiv 2412.12491 ↑470

[378] Larabel M. *AVX-512 performance with 256-bit vs. 512-bit data path for AMD EPYC™ 9005 CPUs.*– 2024 (Accessed 12.10.2024). URL ↑470

[379] *OpenFOAM on 5th Gen AMD EPYC™ Processors.*– 2024 (Accessed 1.12.2024). URL ↑471

[380] Siddiqi S., Kern R., Boehm M. *SAGA: a scalable framework for optimizing data cleaning pipelines for machine learning applications* // Proceedings of the ACM on Management of Data.– 2023.– Vol. **1**.– No. 3.– id. 218.– 26 pp. doi ↑476

[381] Shen Y., Ren X., Lu Yu., Jiang H., Xu H., Peng D., Li Y., Zhang W., Cui B. *Rover: An online Spark SQL tuning service via generalized transfer learning // KDD '23: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA, 6–10 August 2023), New York: ACM.– 2023.– ISBN 979-8-4007-0103-0.– Pp. 4800–4812. ↑476

[382] Li Y., Jiang H., Shen Y., Fang Y., Yang X., Huang D., Zhang X., Zhang W., Zhang C., Chen P., Cui B. *Towards general and efficient online tuning for spark.*– 2023.– 14 pp. arXiv 2309.01901 ↑476

[383] Xin J., Hwang K., Yu Z. *LOCAT: Low-overhead online configuration auto-tuning of Spark SQL applications // SIGMOD '22: Proceedings of the 2022 International Conference on Management of Data* (Philadelphia, PA, USA, 12–17 June 2022), New York: ACM.– 2022.– ISBN 978-1-4503-9249-5.– Pp. 674–684. ↑476

[384] Perera N., Sarker A. K., Staylor M., von Laszewski G., Shan K., Kamburugamuve S., Widanage C., Abeykoon V., Kanewela T. A., Fox G. *In-depth analysis on parallel processing patterns for high-performance Dataframes //* Future Generation Computer Systems.– 2023.– Vol. **149**.– Pp. 250–264. ↑476

[385] Özdil U. E., Ayvaz S. *An experimental and comparative benchmark study examining resource utilization in managed Hadoop context //* Cluster Computing.– 2023.– Vol. **26**.– No. 3.– Pp. 1891–1915. ↑476

[386] Yuan Y., Salmi M. F., Huai Y., Wang K., Lee R., Zhang X. *Spark-GPU: An accelerated in-memory data processing engine on clusters // 2016 IEEE International Conference on Big Data (Big Data)* (Washington, DC, USA, 05–08 December 2016).– IEEE.– 2016.– ISBN 978-1-4673-9005-7.– Pp. 273–283. ↑476

[387] Li P., Luo Y., Zhang N., Cao Y. *HeteroSpark: A heterogeneous CPU/GPU Spark platform for machine learning algorithms // 2015 IEEE International Conference on Networking, Architecture and Storage (NAS).*– IEEE.– 2015.– ISBN 9781467378901.– Pp. 347–348. ↑476

[388] Malakar R., Vydyanathan N. *A CUDA-enabled Hadoop cluster for fast distributed image processing // 2013 National Conference on Parallel Computing Technologies (PARCOMPTECH)* (Bangalore, India, 21–23 February 2013).– IEEE.– 2013.– ISBN 9781479915903.– Pp. 1–5. ↑476

[389] Abbasi A., Khunjush F., Azimi R. *A preliminary study of incorporating GPUs in the Hadoop framework // The 16th CSI International Symposium on Computer Architecture and Digital Systems (CADS 2012)* (Shiraz, Fars, Iran, 02–03 May 2012).– IEEE.– 2012.– ISBN 978-1-4673-1481-7.– Pp. 178–185. ↑476

[390] Azeem M., Abualsoud B. M., Priyadarshana D. *Mobile Big Data analytics using deep learning and Apache Spark //* Mesopotamian Journal of Big Data.– 2023.– Vol. **2023**.– Pp. 16–28. ↑476

[391] Wu R., Wang Y., Kutscher D. *Affordable HPC: leveraging small clusters for big data and graph computing.*– 2024.– 9 pp. arXiv 2408.15568 ↑476, 479

[392] Prichard R., Strasser W. *A Novel HPC scaling optimization methodology* // *Proceeding of 7th Thermal and Fluids Engineering Conference (TFEC)* (Begellhouse, Las Vegas, NV, USA, 15–18 May, 2022).– 2022.– ISBN 978-1-56700-544-8.– Pp. 183–192. doi ↑476

[393] Rogowski M., Dalcin L., Parsani M., Keyes D. E. *Performance analysis of relaxation Runge–Kutta methods* // The International Journal of High Performance Computing Applications.– 2022.– Vol. **36**.– No. 4.– Pp. 524–542. doi ↑477

[394] Che Y., Xu C., Fang J., Wang Y., Wang Z. *Realistic performance characterization of CFD applications on Intel Many Integrated Core architecture* // The Computer Journal.– 2015.– Vol. **58**.– No. 12.– Pp. 3279–3294. doi ↑477

[395] Prichard R., Strasser W. *When fewer cores is faster: a parametric study of undersubscription in high-performance computing* // Cluster Computing.– 2024.– Vol. **27**.– Pp. 9123–9136. doi ↑477

[396] Hammond S., Vaughan C., Hughes C. *Evaluating the Intel Skylake Xeon processor for HPC workloads* // *2018 International Conference on High Performance Computing & Simulation (HPCS)* (Orleans, France, 16–20 July, 2018).– IEEE.– 2018.– ISBN 9781538678800.– Pp. 342–349. doi ↑477

[397] Zhang T., Shirzad Sh., Wagle B., Lemoine A. S., Diehl P., Kaiser H. *Supporting OpenMP 5.0 Tasks in hpxMP – a study of an OpenMP implementation within Task Based Runtime Systems.*– 2020.– 12 pp. arXiv 2002.07970 ↑477

[398] Prichard R., Strasser W. *An intuitive multi-objective, multi-variable high-performance computing optimization methodology* // International Journal of Computational Fluid Dynamics.– 2024.– Vol. **38**.– No. 8–9.– Pp. 610–616. doi ↑478

[399] Chang J., Lu K., Guo Y., Wang Y., Zhao Zh., Huang L., Zhou H., Wang Y., Lei F., Zhang B. *A survey of compute nodes with 100 TFLOPS and beyond for supercomputers* // CCF Transactions on High Performance Computing.– 2024.– Vol. **6**.– No. 3.– Pp. 243–262. doi ↑478, 479

[400] *HPC6, supercomputer at the service of energy* (Accessed 20.04.2025). URL ↑

[401] *GPU nodes — LUMI-G* (Accessed 18.04.2025). URL ↑

[402] *Perlmutter architecture* (Accessed 18.04.2025). URL ↑

[403] *HPE Cray Supercomputing EX QuickSpecs v.19.*– 2024 (Accessed 19.04.2025).– 19 pp. URL ↑480, 481

[404] Fusco L., Khalilov M., Chrapek M., Chukkapalli G., Schulthess Th., Hoefler T. *Understanding data movement in tightly coupled heterogeneous systems: A case study with the Grace Hopper superchip.*– 2024.– 12 pp. arXiv 2408.11556 ↑481

[405] *AMD Instinct MI300A APU Datasheet.*– 2023 (Accessed 21.01.2025).– 2 pp. URL ↑481

(*) Сайт закрыт для российских пользователей (*прим. ред*);

(**) Источник недоступен редакции журнала (*прим. ред*).

Приложение. Используемые в статье сокращения

Здесь приводятся только некоторые, не самые крайне широко используемые в соответствующей научной литературе сокращения (с точки зрения автора), которые применяются в тексте статьи в самых разных разделах, а не только в ориентированных на конкретные модели процессоров.

Общие сокращения

1P one-processor . . *2.1, 2.2, 2.3, 2.4, 3.2, 4.1, 4.2, 4.3, 4.4, 6, 7*

2P two-processor *Введение, 1.1, 1.3, 2.1, 2.2, 2.3, 2.4, 3, 3.1, 3.2, 4.1, 4.2, 4.3, 4.4, 6, 7*

4P four-processor *2.4*

8P eight-processor *2.4*

1DPC One DIMM per channel *2.2, 2.4, 3.1, 3.3, 4.1*

2DPC Two DIMM per channel *2.2, 2.4, 3.1, 3.3, 4.1*

ACPI Advanced Configuration and Power Interface . . . *1.2, 3.2*

FMA Fused multiply-add *2.1, 2.4, 3*

NPB NAS Parallel Benchmarks *2.4, 4.1, 4.3, 4.4, 6*

PTS Phoronix Test Suite . ***Введение**, 2.4, 4.1, 4.2, 4.3, 4.4, 6, 7*

SKU Stock Keeping Unit (стандартное во всем мире сокращение для кода любого товара) *2.2, 3, 3.1, 3.2, 3.3*

SMT Simultaneous multithreading *Введение, 2, 2.2, 2.3, 2.4, 3, 3.1, 4.1*

Сокращения для аппаратных средств AMD:

CCD Core Complex Die *2.2, 2.3, 2.4, 3*

CCX Core Complex *2.1, 2.2, 2.4*

GMI Global Memory Interface *2.2*

IF Infinity Fabric *2.2, 2.3, 3*

IOD I/O Die (теперь применяется также в Xeon 6) *2.2, 2.3, 6, 7*

NPS Nodes Per Socket *2.2, 2.3, 2.4, 7*

xGMI eXternal Global Memory Interconnect *2.2, 2.3*

Сокращения для аппаратных средств Intel:

1LM one-level memory *3.2*

2LM two-level memory *3.2*

AMX Advanced Matrix Extensions *1.3, 2.1, 3.1, 3.2, 3.3, 4.1, 4.2, 6, 7*

EMR Emerald Rapids *Введение, 2.4, 3, 3.1, 3.3, 4, 4.1, 4.2, 4.3, 4.4, 6, 7*

ICL Ice Lake . . . . . *Введение, 1.1, 2.1, 2.4, 3, 3.1, 4.1, 4.3, 4.4*

OPM Optimized Power Mode *3.1, 4.1, 4.2*

SNC 4 sub-NUMA clustering *3, 3.1, 3.2, 3.3, 4.1, 4.4*

SPR Sapphire Rapids *Введение, 1.1, 1.2, 1.3, 2.1, 2.4, 3, 3.1, 3.2, 3.3, 4, 4.1, 4.2, 4.3, 4.4, 6, 7*

UPI Ultra Path Interconnect *3, 3.1, 3.2, 3.3, 6*

Аналоги терминов Intel и AMD:

AMD	*Intel*
NPS	SNC
xGMI	UPI

Поступила в редакцию 31.05.2025;
одобрена после рецензирования 14.10.2023;
принята к публикации 10.11.2025;
опубликована онлайн 15.01.2026.

Информация об авторе:

Михаил Борисович Кузьминский

старший научный сотрудник лаборатории компьютерного обеспечения химических исследований, кандидат химических наук ИОХ РАН. Научные интересы – высокопроизводительные вычисления, аппаратура ЭВМ, вычислительная химия.

0000-0002-3944-8203

e-mail: *kus@free.net*

Декларация об отсутствии личной заинтересованности: *благополучие автора не зависит от результатов исследования.*