White Paper

3rd Gen Intel Xeon Scalable Platform

Cryptography Processing with 3rd Gen Intel Xeon Scalable Processors

Cryptographic operations are amongst the most compute intensive and critical operations applied to data as it is stored, moved, and processed. Comprehending Intel's cryptography processing acceleration is essential to optimizing overall platform, workload, and service performance.

Authors

Ryan D. Saffores

Intel Data Platforms
Engineering Manager

Shruthi Venugopal

Intel Data Platforms
Silicon Architect Engineer

Table of Contents

Introduction

Reports show that approximately 40% of organizations have adopted encryption in the public cloud by 2019.[1] Another trend, due to increased threats, is the need for stronger encryption algorithms and larger keys – all this combined with the fact that data and traffic is increasing exponentially and will substantially increase the compute cycles spent on cryptography processing.[2]

New 3rd Gen Intel Xeon Scalable processors, codenamed Ice Lake-SP, were launched April 6, 2021 and have new capabilities to enhance cryptographic operations, known as Intel Crypto Acceleration. These processors are supported in one-and-two socket server configurations and are the first generation of Intel Xeon Scalable processors that support two Intel Advanced Vector Extensions 512 (Intel AVX-512) Fuse Multiply Add (FMAs) across all Intel Xeon Platinum, Gold and Silver processors, resulting in enhanced crypto performance across all 3rd Gen Intel Xeon Scalable processors. In addition, these processors are the first to support new Intel AVX-512 “light” and Intel AVX-512 “heavy” ISA instructions.

These crypto new instructions help to:

- Improve performance and SLA, supported by faster cryptographic algorithms
- Enable implementation of stronger encryption protocols (larger key sizes, stronger algorithms) without compromising performance
- Reduce compute cycles allocated to cryptography processing

Intel is implementing support for the most common cryptography algorithms in low-level performance libraries and continues to work with the broader ecosystem to enable support in commonly used software such as OpenSSL, BoringSSL, Linux Kernel, Java, Golang and Kubernetes.

In this paper we will give the details behind our new Intel Crypto Acceleration and our new crypto instructions, along with their expected usages. We will also dive into some specific use cases you may see with frequency scaling/transitions, how it has improved on our 3rd Gen Intel Xeon Scalable processors as well as share some data to demonstrate the performance improvements and why frequency reductions should not materially impact performance.

3rd Gen Intel Xeon Scalable Processors, Crypto Acceleration Instructions

Instructions	CPUID Feature Flags	Application to Cryptography	Common Usages
VPMADD52HUQ VPMADD52LUQ	AVX512_IFMA AVX512VL	Supports efficient big number multiplication, squaring and accumulation operations used in the computation of public key cryptographic ciphers such as RSA, ECDSA, and ECDHE.	Authentication and Key Exchange algorithms used in cryptographic protocols such as Transport Layer Security (TLS) and Secure Shell (SSH).
VAESENC VAESDEC VAESENCLAST VAESDECLAST VAESIMC VAESKEYGENASSIST	VAES AVX512F AVX512VL	Supports vectorized Advanced Encryption Standard (AES) cipher implementations processing up to four 128-bit blocks per instruction resulting in improved cipher throughput.	AES is the de facto cipher for strong and efficient bulk encryption. Often used in TLS cipher suites, but also found in full disk encryption, IPSec, and many other secure messaging protocols.
VPCLMULQDQ	VPCLMULQDQ AVX512F AVX512VL	Supports vectorized carry-less multiplication for efficient computation in Galois Hash (GHASH) message digest.	GHASH is used for high performance message authentication and is commonly used in conjunction with AES to make up the AES-GCM cipher. AES-GCM is arguably the most widely adopted/deployed bulk encryption cipher used in TLS and IPSec today.
SHA256 SHA256MSG2MSG1 SHA256RNDS2	SHA	Supports efficient computation of Secure Hash Algorithm 2 (SHA) 256-bit message digests.	SHA-256 usage can be found in protocols such as TLS, IPSec, SSH, but also can be used wherever data/message integrity is required.

Figure 1 . 3rd Gen Intel Xeon Scalable Processors, Crypto ISA and Usages

Application Performance

The following examples show the performance gain, frequency reduction and percentage of cycles spent at each license level for a variety of workloads that use the new crypto instructions. In our experience the performance gain on 3rd Gen Xeon Scalable due to the new crypto instructions outweighs the minimal performance loss of 0-2 bin reduction in frequency for the rest of the workload that previously ran at 100% Intel SSE frequency.

In the tables below “SW Enc” denotes running in the default mode with SW Encryption that does not utilize the new crypto instructions and “HW Enc” shows where we’re using the new instructions.

NGINX

For this NGINX benchmark we're tracking bulk web server throughput by sending a 10MB file from Web Server to N-Number of Web Client(s). It uses Key Exchange + Certificate Authentication + Packet Encryption + Packet Authentication and our KPI is Connection-Per-Second (CPS). The system configuration is available in the Appendix.

Test	Score	Avg CPU Freq	% Intel SSE	% Intel AVX2	% Intel AVX-512
NGINX RSA 2k SW Enc	31,257.68 Connections/sec	2.19 GHz	100%	0%	0%
NGINX RSA 2k HW Enc	96,101.60 Connections/sec	1.96 GHz	23.72%	0.47%	75.81%

NGINX RSA 2K performance improvement of 3.07x with the new 3rd Gen Intel Xeon Scalable crypto instructions

HAProxy

HAProxy is free, open-source software that provides a high availability load balancer and proxy server for TCP and HTTP-based applications that spreads requests across multiple servers. It is written in C language and has a reputation for being fast and efficient (in terms of processor and memory usage). To enable HAProxy to use the new crypto instructions the OpenSSL Intel QuickAssist Technology (Intel QAT) software engine was used, and no changes required on the HA-Proxy build to exercise the RSA multi-buffer feature. The system configuration is available in the Appendix.

Test	Score	Avg CPU Freq	% Intel SSE	% Intel AVX2	% Intel AVX-512
HAProxy SW Enc	51,659 Req/sec	2.89 GHz	100%	N/A	N/A
HAProxy HW Enc	125,129 Req/sec	2.69 GHz	51.27%	12.01%	36.63%

HAProxy performance improvement of 2.42x with the new 3rd Gen Intel Xeon Scalable crypto instructions

WordPress

This workload is based on the open source "OSS-Performance" workload. It uses Siege as the load generator, PHP for server-side application logic, NGINX web server and MariaDB (MySQL) as the database. The system configuration is available in the Appendix.

Test	Score	Avg CPU Freq	% Intel SSE	% Intel AVX2	% Intel AVX-512
TLS 1.2 SW Enc	6017.70 Tx/sec	2.93 GHz	100%	0%	0%
TLS 1.2 HW Enc	6515.73 Tx/sec	2.83 GHz	51.72%	21.11%	27.17%

TLS 1.2 performance improvement of 1.08x with the new 3rd Gen Xeon Scalable crypto instructions

TLS 1.3 SW Enc	5893.19 Tx/sec	2.93 GHz	100%	0%	0%
TLS 1.3 HW Enc	6882.59 Tx/sec	2.84 GHz	50.58%	28.35%	21.06%

TLS 1.3 performance improvement of 1.16x with the new 3rd Gen Intel Xeon Scalable crypto instructions

3rd Gen Intel Xeon Scalable Processors, Crypto Acceleration Instructions

When using these new instructions, they run at either the Intel Advanced Vector Extensions 2 (Intel AVX2) or Intel AVX-512 frequency level depending on the instruction. As a use case cryptography instruction usage is often intermittent. Sporadic use of the Intel AVX family of instructions can have a performance impact on applications that run mostly at the Intel Streaming SIMD Extensions (Intel SSE) frequency for a few reasons that we'll detail more in this paper. At a high level you may run into the following behaviors:

1. Core frequency "bin" reduction. When crypto instructions are executed, the frequency on the core executing the instruction may be reduced to Intel AVX2 or Intel AVX-512 base frequencies. After the instruction is executed, it may take milliseconds for the frequency to increase back Intel SSE base frequency. Effectively, a small number of Intel AVX family of instructions can result in a frequency impact for a short time. Starting with 3rd Gen Intel Xeon Scalable processors, we try to reduce the "depth" (number of bins lost) of this frequency reduction and this is an improvement compared to previous generations of Xeon processors.
2. During the frequency transition from low (Intel SSE) to high (Intel AVX2/Intel AVX-512) there is transient core throttling during the transition for some microseconds. This can result in a performance degradation for very short periods (10's of usec) during the transition period.
3. While executing Intel AVX2 Heavy code (which from now on will be referred to as AVX2H) on a core, the concurrent execution of Intel AVX-512 Light code (which from now on will be referred to as AVX512L) on that core's attached hyperthread for some time can cause the system to reduce frequency to the AVX-512 Heavy (which from now on will be referred to as AVX512H) turbo frequency. On 3rd Gen Intel Xeon Scalable processors, we have improved this by increasing the amount of time you have to run the two bits of code concurrently to trigger this frequency reduction.
4. Conditional usage of Intel AVX-512 versus Intel AVX2 instructions may result in performance degradation due to mis-speculation into the Intel AVX-512 branch. This conditional execution of Intel AVX-512 v/s Intel AVX2 can be still supported but with a simple workaround discussed in this paper.

Intel Advanced Vector Extensions (Intel AVX) Behavior

Architecture Background of Intel AVX Instructions

To understand the details around Intel AVX license usage with the crypto instructions on the 3rd Gen Intel Xeon Scalable processors, this section will give some background on Intel AVX terminology and the frequency behavior behind Intel AVX instructions.

Intel AVX instructions support larger vector operations and can provide significant performance improvement over Intel SSE.

Intel Xeon E3 v3 processors, formerly known as Haswell, added support for Intel AVX2 instruction set which allows up to 256-bit vector operations. Intel Xeon Scalable processors, formerly known as Skylake, added support for Intel AVX-512 supporting up to 512-bit vector operations.

Some instructions, especially those with wide operands and complex computational operations, can have a significant power draw, much higher than that of baseline Intel SSE instructions. Rather than burden all workloads to account for this, instructions are grouped into classes, with some classes of instructions are restricted to lower frequencies to fit within processor constraints.

Intel Turbo Boost Technology

Intel Turbo Boost Technology automatically and opportunistically enables processor cores to run faster than the configured TDP frequency if the processor is operating below power, current, and temperature specification limits. The availability and amount of turbo frequency depends on the workload and operating environment, specifically the following factors:

- Processor SKU
- Number of Active Cores in C0 State
- Type of Workload
- Estimated current consumption
- Estimated power consumption
- Processor temperature

Core License Levels

At a high level, "Core License Level" refers to the type and bit-width of the instruction. Core License Levels are bucketed and mapped to voltage/frequency (V/F) Curves and Turbo Ratio Limits by processor.

Width	License Level
64-128L	0
128H	1
256L	2
256H	3
512L	4
512H	5

Figure 2 . Core license level mapping

License Instruction Mapping

The following table shows the Frequency Levels Mapping for different instruction types across 3rd Gen Intel Xeon Scalable processors.

Core License Level	Name	Width	Max Turbo Schedule	Instructions
1	Non-AVX	128	SSE	Intel AVX Scalar, Intel AVX 128, Intel SSE, Everything Else
2	AVX2 Light	256	SSE	Intel AVX2 w/out FP or INT MUL/FMA
3	AVX2 Heavy	256	AVX2	Intel AVX2 FP + INT MUL/FMA
4	AVX-512 Light	512	AVX2	Intel AVX-512 w/out FP or INT MUL/FMA (This includes V-CLMUL, V-AES)
5	AVX-512 Heavy	512	AVX-512	Intel AVX-512 FP + INT MUL/FMA (This includes VPMADD512)

Figure 3 . License instruction mapping for 3rd Gen Intel Xeon Scalable processors

Turbo Ratio Limits

The highest all core turbo frequency (P0nMax) that an instruction set can run at depends on the Max Real-Workload Dynamic Power Draw (Cdyn) that the instruction set can potentially hit. This principle is used to map core license levels to Turbo Ratio Limits (TRL). This mapping is done based on the width and type of instruction set. The Core License Levels are classified into "similar Cdyn" buckets and each bucket is mapped to the respective TRL. As the principle is based on "similar cdyn" buckets, the Heavy instructions of a particular width are mapped to the same Cdyn bucket as the Light instructions of the next width size.

Frequency Improvements Between 2nd and 3rd Gen Intel Xeon Scalable Processors

The following charts demonstrate the improvement in the "depth" of all core turbo frequency reduction comparing our 2nd Gen Intel Xeon Scalable processors, codenamed Cascade Lake-SP, to 3rd Gen Intel Xeon Scalable processors, codenamed Ice Lake-SP. Our Intel AVX2 frequencies have drastically improved on new 3rd Gen Xeon Scalable, with most SKUs having no reduction compared to an average of approximately 4 bins reduced on second generation (note 1 bin = 100 MHz). From the chart below, over 55% of third generation SKUs have no frequency reduction.

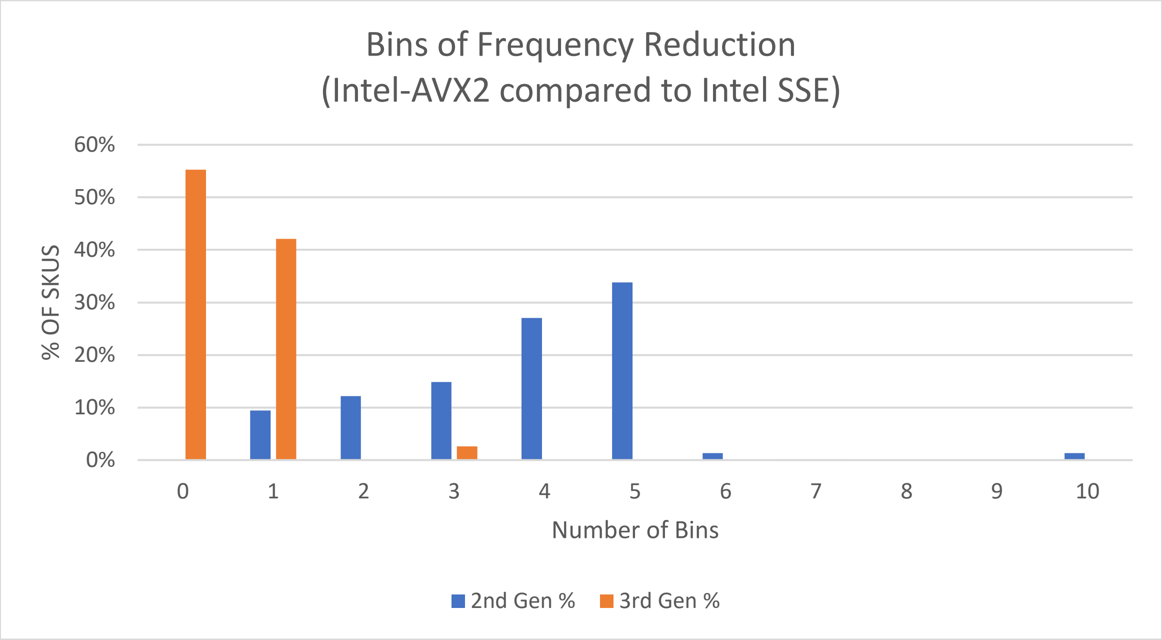


Figure 4 . Comparing bins of frequency reduction between Intel SSE and Intel AVX2 for 2nd and 3rd Gen Intel Xeon Scalable processors

Intel AVX-512 has also improved shifting the bulk of the reduction from 8-16 bins on second generation to 7 or less on third generation, with 50% of third generation SKUs only reducing 2 bins (68% of SKUs reducing 2 bins or less).

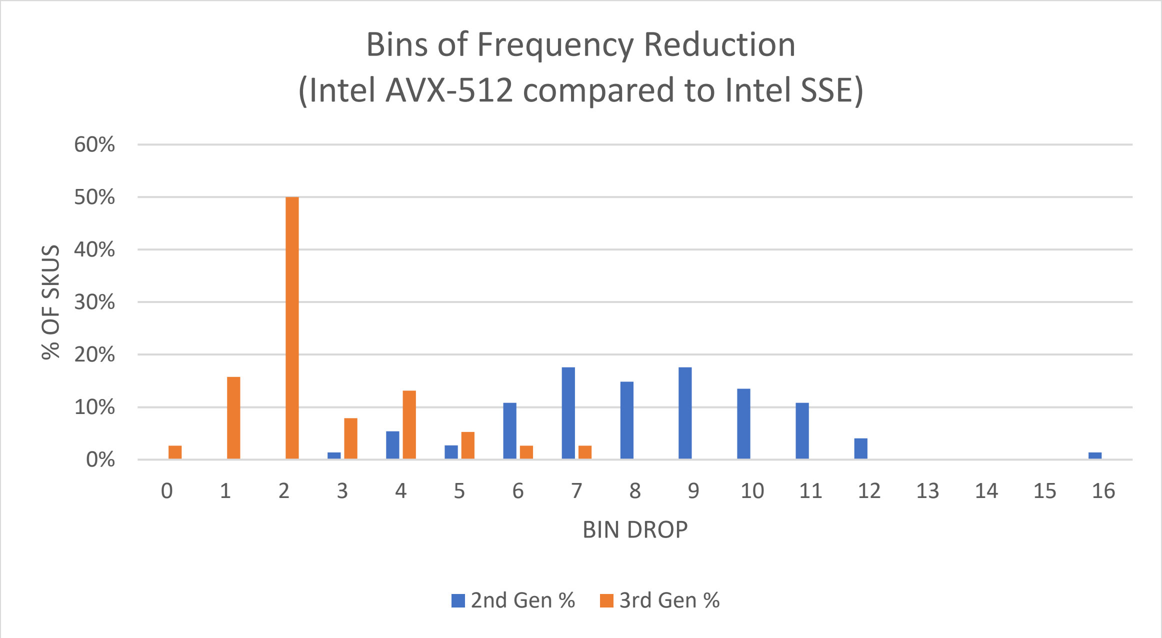


Figure 5 . Comparing bins of frequency reduction between Intel SSE and Intel AVX512 for 2nd and 3rd Gen Intel Xeon Scalable processors

The following table shows where the new crypto instructions map and the expected frequency reduction on 3rd Gen Xeon Scalable. We've greatly improved the amount of frequency reduced with our Intel AVX2 and Intel AVX-512 instructions on 3rd Gen Xeon Scalable compared to previous generations.

Width	Core License	Crypto ISA	Max 3rd Gen Intel Xeon Scalable Frequency Reduction
64- 128	1	SHA-NI	0 bins
256L	2	SHA-NI	0 bins
256H	3	SHA-NI	2 bins
512L	4	VAES, VCLMUL, VPCLMULQDQ, GFNI	2 bins
512H	5	VPMADD52	7 bins

Figure 6 . Mapping the new crypto instructions to their license levels and comparing to the max frequency reduction expected.

License Transition Latencies

As a general concept, transitioning from lower core license (Intel SSE) to a higher core license (Intel AVX2 or Intel AVX-512) triggers a transient frequency reduction in the core until the higher license is granted. Whereas, transitioning from higher core license (Intel AVX-512) to a lower license (Intel SSE or Intel AVX2) is based on hysteresis so the frequency increase will lag behind the introduction of the new instructions. It should be noted that transitioning from higher core license to a lower core license no longer triggers any transient throttling in the core.

3rd Gen Intel Xeon Scalable Processors License Transition Latencies

License Transition Latencies	
Low -> High	High -> Low
~0.02- 0.2 ms	~1 -4 ms

Figure 7 . License transitions latencies on 3rd Gen Intel Xeon Scalable processors

We've measured these latencies by running a test that starts running 100% Intel SSE code and then injects code at the AVX512L license level after n usec. After some time, it stops running the higher license code and reverts back to the pure Intel SSE code to show the high to low transition. In Figure 8 you can see a comparison of license transition latencies between 2nd and 3rd Gen Intel Xeon Scalable processors.

See next page for Figure 8.

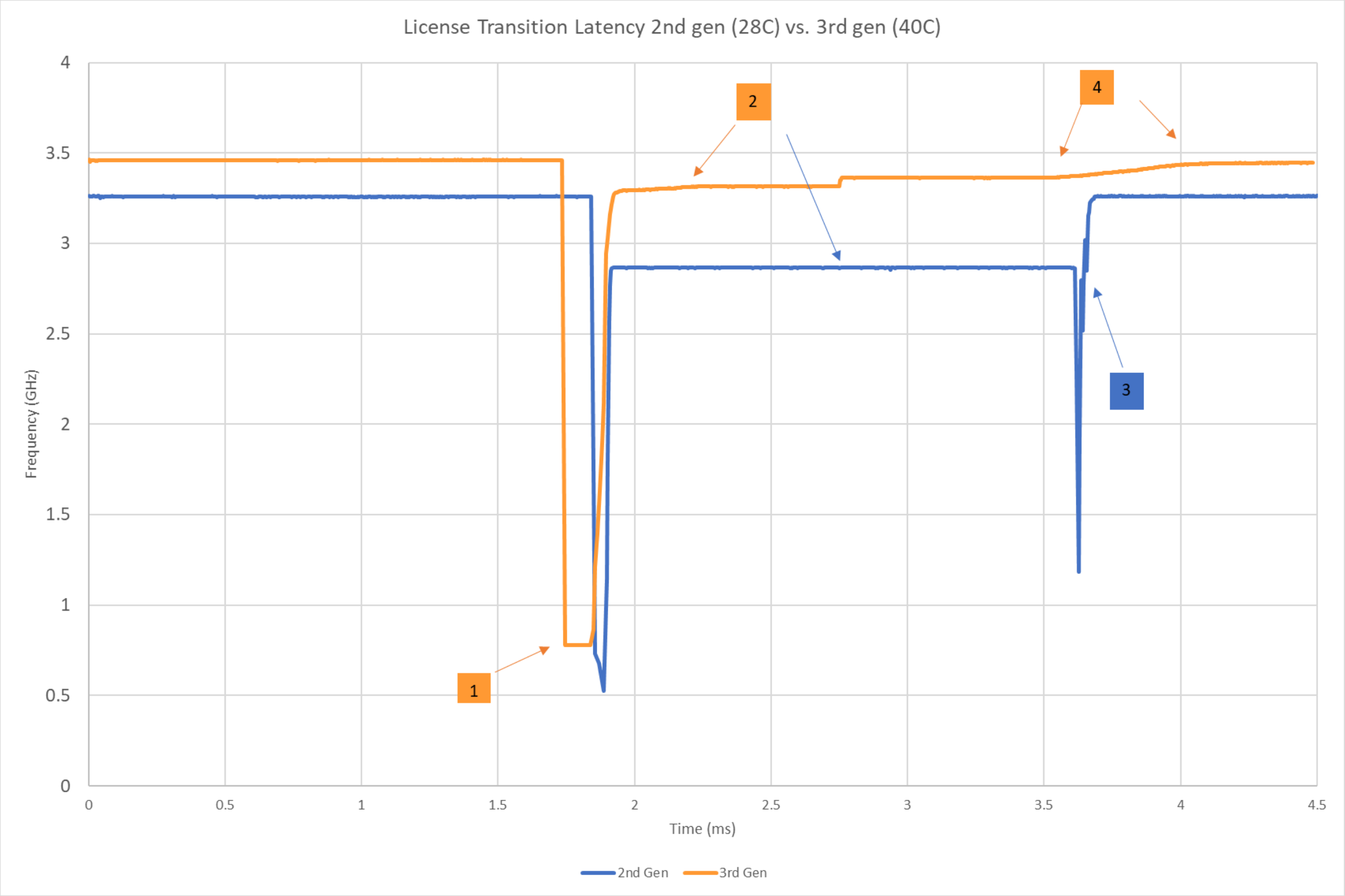


Figure 8 . Comparing the license enter/exit latency between 2nd and 3rd Gen Intel Xeon Scalable processors

1. Longer transition from a higher frequency to a lower frequency due to the higher core count on ICX. A comparison at equal core count would show similar characteristics between 2nd Gen and 3rd Gen Xeon Scalable
2. As detailed in the frequency section this demonstrates the improvement in AVX2H between 3rd and 2nd Gen Xeon Scalable. The 3rd Gen Xeon Scalable only reduces 2 bins where the 2nd Gen Xeon Scalable reduces 4 bins.
3. We've made an improvement on our 3rd Gen Xeon Scalable so that lower frequency to higher frequency transitions no longer trigger core throttling
4. Our 3rd Gen Xeon Scalable have a longer, more gradual return to a higher frequency from a lower one.

3rd Gen Intel Xeon Scalable Processors License Transition Latencies

While executing AVX2H code on a core and simultaneously executing AVX512L code on the attached sibling thread when Intel Hyper-Threading Technology (Intel HT Technology) is enabled for some time can cause the system to reduce frequency to the AVX512H turbo frequency. We've demonstrated this behavior with a contrived microbenchmark below.

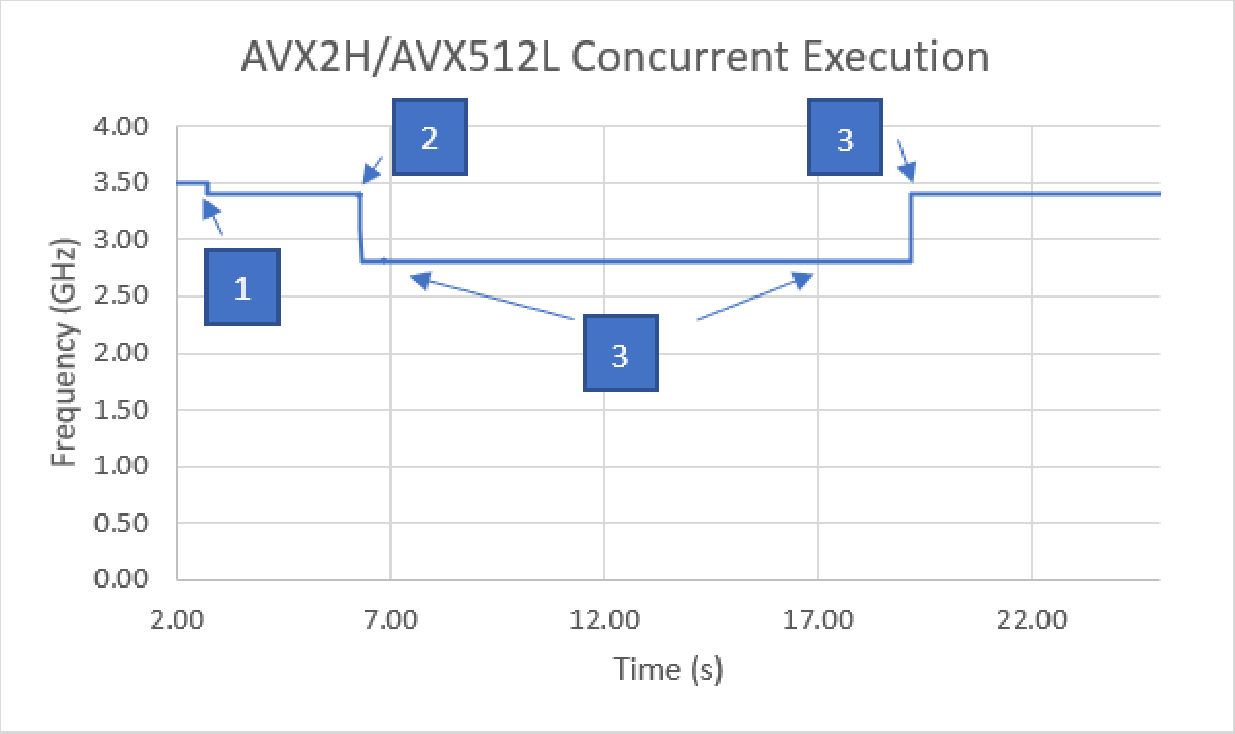


Figure 9 . Frequency reduction due to concurrent AVX2H/AVX512L execution

1. AVX2H instruction runs in thread-0 of the core. Frequency is at Intel AVX2 P0n
2. AVX512L instruction kicks-in in thread-1 of the core
3. The instruction mix AVX2H (tight loop) + AVX512L(in us interval) runs at Intel AVX512 P0n
4. AVX512L instruction stops. Frequency goes back to Intel AVX2 P0n

Minimizing Core Frequency Impacts Using AVX512VL

Intel AVX-512 Vector Length Extension (AVX512VL) allows one to take advantage of Intel AVX-512 features (32 SIMD registers, masking, new instructions) with 128-bit and 256-bit vector widths. While in some cases using 512-bit register width would require lowering the frequency due to the power demand, the use of smaller register widths can often avoid that. Frequency change can have a measured impact on the code running immediately after, or interleaved with, Intel AVX-512 code. Although maximum performance/efficiency is typically gained through the use of 512-bit zmm registers, some algorithms with lower overall fraction of SIMD code may find better performance, on balance, by using narrower than 512-bit register widths to avoid lower frequencies but still taking advantage of other Intel AVX-512 features mentioned above. An example of this optimization technique can be found in the OpenSSL modular exponentiation implementation which substantially improves RSA 2048 signing performance over prior OpenSSL implementations on the same processor hardware. The code makes use of Intel AVX-512 VPMADD52 instructions with 256-bit register width, takes advantage of AVX512VL, and avoids a frequency impact on the calling workload code.

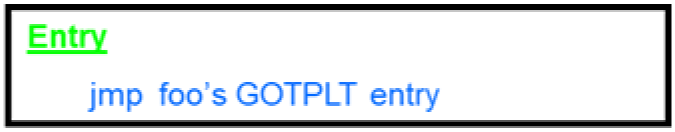
Workaround for Mis-speculation

The best known method (or user interface) to prevent possible mis-speculation issues around executing code such as "`IF () AVX2 VAES ELSE AVX512 VAES`" is to provide an indirection function pointer instead of using if –else constructs. The fundamental idea is we provide an infrastructure in the GNU C library and GNU assembler/linker.

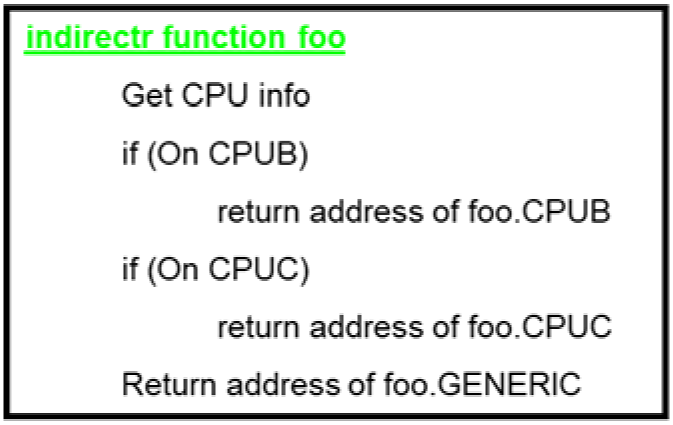
PLT
Indirect jump to foo's GOTPLT entry

GOTPLT
Address of function 'foo'

Let's assume an external function foo is called via its PLT entry. Now the dynamic linker fills the GOTPLT entry of foo with the address of the function returned by indirect function foo. A typical function call would look like as shown below:



```
Entry
    jmp  foo's GOTPLT  entry
```



```
indirectr function foo
        Get CPU info
        if (On CPUB)
                return address of foo.CPUB
        if (On CPUC)
                return address of foo.CPUC
        Return address of foo.GENERIC
```

Studying it Yourself

There are many tools you can use to track average frequency of your system during execution (cat /proc/cpuinfo, lscpu, perf, etc). An example using Linux perf is below (see frequency in bold):

```
perf stat -a
Performance counter stats for 'system wide':

     357081.948497      cpu-clock (msec)             #  111.874 CPUs utilized
               703      context-switches             #    0.002 K/sec
               144      cpu-migrations               #    0.000 K/sec
               206      page-faults                  #    0.001 K/sec
   954,048,762,806      cycles                       #    2.672 GHz
 1,403,375,002,401      instructions                 #    1.47  insn per cycle
   420,629,628,853      branches                     # 1177.964 M/sec
       755,940,960      branch-misses                #    0.18% of all branches
```

To observe the potential for your application behavior with reduced frequency based on license level transitions you can observe the amount of time your app spends at each frequency level using Linux perf with the following command:

```
perf stat -e CORE_POWER.LVL0_TURBO_LICENSE,CORE_POWER.LVL1_TURBO_LICENSE,CORE_POWER.LVL2_TURBO_LICENSE
```

If you run this in conjunction with your application it will track the time spent at each Turbo Ratio Limit. Example of system running 100% Intel SSE instructions:

```
Performance counter stats for 'system wide':
       287,441,999      CORE_POWER.LVL0_TURBO_LICENSE
                 0      CORE_POWER.LVL1_TURBO_LICENSE
                 0      CORE_POWER.LVL2_TURBO_LICENSE
```

Tying this back to our earlier table the below is where these perf counters map:

Width	Perf stat	TRL Mapping
64-126L	CORE_POWER.LVL0_TURBO_LICENSE	0
128H	CORE_POWER.LVL0_TURBO_LICENSE	0
256L	CORE_POWER.LVL0_TURBO_LICENSE	0
256H	CORE_POWER.LVL1_TURBO_LICENSE	1
512L	CORE_POWER.LVL1_TURBO_LICENSE	1
512H	CORE_POWER.LVL2_TURBO_LICENSE	2

Closing

Sporadic use of Intel AVX2/Intel AVX-512 instructions has not been a recommended practice on past generations of Intel Xeon processors. On 3rd Gen Intel Xeon Scalable processors, the changes we have made in frequency behavior along with performance improvements of new Intel Crypto Acceleration instructions should far outweigh any potential performance side effects due to reduced frequency. This is especially true when using the new crypto instructions that work at the Intel AVX2 level and benefit from improved Intel AVX2 frequencies on the 3rd Gen Intel Xeon Scalable processors.

Learn More

Please visit **intel.com/avx512** and **software.intel.com** to access customer stories, resources, and the developer community supporting the capabilities of Intel AVX-512 and Intel Crypto Acceleration.

Appendix

Appendix 1: WordPress and HammerDB PostgreSQL Configuration

CPU SKU	8380
Wordpress	5.2
Siege	v2.78: built-in support for HTTPS, if libssl is detected on the system
PHP	PHP v7.3.23
MariaDB	v10.3.22
NGINX	v1.18 with https://github.com/intel/asynch_mode_nginx
OpenSSL	1.1.1f with Intel QAT Engine v0.6.1
Ciphers	TLS_AES_256_GCM_SHA384 (TLSv1.3)

Appendix 2: NGINX Configuration

CPU SKU	6338N @ 2.2GHz
NGINX Version	Intel Async NGINX v0.4.3
OpenSSL	OpenSSL 1.1.h
IPP-Crypto	2020 Update 3 https://github.com/intel/ipp-crypto.git
Intel QAT Engine for OpenSSL	v0.6.4 https://github.com/intel/QAT_Engine.git

Appendix 3: HAProxy Configuration

CPU SKU	6330
HAProxy Version	2.0.13
OpenSSL	OpenSSL 1.1.f
Intel Multi-Buffer Crypto for OpenSSL	v0.55 https://github.com/intel/intel-ipsec-mb.git
Intel QAT Engine for OpenSSL	v0.6.4.2 (available under NDA upon request)

References

1. https://www.techrepublic.com/article/43-of-enterprises-have-adopted-an-encryption-strategy/

2. https://aws.amazon.com/blogs/security/post-quantum-tls-now-supported-in-aws-kms/

Performance varies by use, configuration and other factors. Learn more at www.intel.com/performanceindex

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Intel contributes to the development of benchmarks by participating in, sponsoring, and/or contributing technical support to various benchmarking groups, including the BenchmarkXPRT

Your costs and results may vary.

Intel technologies may require enabled hardware, software or service activation.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

All product plans and roadmaps are subject to change without notice.