WP512 (v1.0) May 21, 2019

Accelerating Cryptographic Performance on the Zynq UltraScale+ MPSoC

The Zynq® UltraScale+™ MPSoC's embedded cryptographic accelerator enables system architects to greatly increase cryptographic performance—by as much as 10,000% or more—compared to software-only solutions.

ABSTRACT

The Zynq UltraScale+ MPSoC features embedded cryptographic cores[Ref 1] for AES-GCM-256 bit, SHA-3/384, and RSA, as well as Arm® v8 cryptographic extensions[Ref 2] within the Arm Cortex®-A53 processor, providing accelerated cryptographic performance.

The performance of the embedded cryptographic accelerators is described under two software architectures, running on a real-time operating system (RTOS) as well as in user space on Linux. In each scenario, the cryptographic accelerator's performance is compared to using a software-only cryptographic solution. These performance measurements show how system architects can maximize cryptographic performance depending on their needs.

Introduction

This white paper illustrates:

- The performance of software running on the Arm Cortex-A53 processor, leveraging the built-in cryptographic accelerators in the Zynq UltraScale+ MPSoC's Configuration Security Unit (CSU)
- The performance of the equivalent software algorithm running on the Arm Cortex-A53 processor, but leveraging the Arm v8 cryptographic extensions

Both results are compared to the native Arm Cortex-A53 software solution.

The Zynq UltraScale+ MPSoC accelerators located in the CSU contain the embedded cryptographic cores for AES-GCM 256-bit encryption and decryption, SHA-3/384 hashing, and both public and private RSA operations with a key size of up to 4,096 bits. Although the default clock for the CSU is the Zynq UltraScale+ MPSoC's internal system oscillator running at 180MHz[Ref 1], the clock was changed for these tests to run off the low power domain's phase-locked loop (PLL), set at 375MHz, to achieve better performance. The cryptographic extensions available in the Arm v8 core accelerate AES, SHA-1, SHA-2, and CRC-32 operations, but they do not support acceleration of any RSA or SHA-3 operations.

Performance measurements for all tests were run on the Arm Cortex-A53 processor in the Zynq UltraScale+ MPSoC, and utilized the wolfSSL[Ref 3] built-in benchmarking software version 3.12.0 running out of external dual data rate (DDR) memory. WolfSSL is able to leverage a range of hardware for cryptographic functions during cryptographic performance measurements. On the Zynq UltraScale+ MPSoC, this includes the embedded cryptographic core, Arm v8 cryptographic extensions, and software-only algorithms. Data sets of 16 bytes up to 15,888 bytes were used to measure the performance of the AES-GCM 256-bit and SHA3/384 cryptographic algorithms, while RSA-2048 and RSA-4096 were used to measure RSA performance.

wolfSSL on FreeRTOS

WolfSSL version 3.12.0 was ported to FreeRTOS on Xilinx SDK 2017.1 for the Zynq UltraScale+ MPSoC to benchmark the performance of the software-only solution, Arm v8 cryptographic extensions, and the Zynq UltraScale+ MPSoC's embedded cryptographic core for the AES-GCM 256-bit, SHA3/384, RSA-2048, and RSA-4096 algorithms. CSU DMA caching and endian byte-swapping was disabled in the XilSecure libraries[Ref 4] for the wolfSSL FreeRTOS port to maximize the embedded cryptographic performance. Data block sizes of 16 bytes, 528 bytes, 1,024 bytes, 4,112 bytes, 7,696 bytes, and 15,888 bytes were used to measure the throughput of the AES-GCM 256-bit and SHA3/384 algorithms. RSA-2048 and RSA-4096 public and private operations were performed to measure the average amount of time needed for each operation. For the RSA operations, the pre-calculated exponential value was not used.

Results

Table 1 shows the results of AES-GCM 256-bit encryption running the wolfSSL benchmarks under FreeRTOS. This table highlights in **bold** the fastest performing cryptographic extension and shows that the Arm v8 cryptographic extensions outperform the Zynq UltraScale+ MPSoC's embedded cryptographic core for data blocks of less than or equal to 1,024 bytes, while the MPSoC's crypto core outperforms the Arm v8 cryptographic extensions for data blocks greater than 1,024 bytes.

Table 1: **FreeRTOS Throughput, AES-GCM Encryption**

Block Size	FreeRTOS Operation Throughput: AES-GCM 256-bit Encryption (MB/s)		
	Software Only	Arm v8 Crypto Extensions	Zynq UltraScale+ MPSoC Crypto Hardware
16	3.785	**53.041**	4.012
528	7.52	**278.156**	122.285
1,024	7.634	**297.949**	219.531
4,112	7.73	315.976	**555.58**
7,696	7.743	318.9	**691.379**
15,888	7.746	318.949	**844.344**

Table 2 shows the results of AES-GCM 256-bit decryption running the wolfSSL benchmarks under FreeRTOS. This table highlights in **bold** the fastest performing cryptographic extension and shows that the Arm v8 cryptographic extensions outperform the Zynq UltraScale+ MPSoC's embedded cryptographic core for data blocks of less than or equal to 1,024 bytes, while the MPSoC crypto core outperforms the Arm v8 cryptographic extensions for data blocks greater than 1,024 bytes.

Table 2: **FreeRTOS Throughput, AES-GCM Decryption**

Block Size	FreeRTOS Operation Throughput: AES-GCM 256-bit Decryption (MB/s)		
	Software Only	Arm v8 Crypto Extensions	Zynq UltraScale+ MPSoC Crypto Hardware
16	3.756	**43.414**	2.069
528	7.515	**166.307**	59.153
1,024	7.634	**175.415**	100.244
4,112	7.73	183.429	**237.055**
7,696	7.75	184.771	**291.194**
15,888	7.746	184.854	**373.363**

In Table 3, the performance of SHA3/384 reveals that the software-only solution outperforms the Zynq UltraScale+ MPSoC's embedded cryptographic core only for data blocks of 16 bytes.

Table 3: **FreeRTOS Throughput, SHA3/384**

Block Size	FreeRTOS Operation Throughput: SHA3/384 (MB/s)	
	Software Only	Zynq UltraScale+ MPSoC Crypto Hardware
16	**38.502**	17.934
528	55.087	**328.623**
1,024	56.152	**417.92**
4,112	57.295	**592.763**
7,696	57.615	**628.626**
15,888	57.52	**654.565**

Table 4 shows that the Zynq UltraScale+ MPSoC embedded cryptographic core outperforms all software-only based RSA solutions.

Table 4: **FreeRTOS Average RSA Operation Time**

RSA Operation	FreeRTOS Avg. RSA Operation Time: wolfSSL v3.12.0 (ms)	
	Software Only	Zynq UltraScale+ MPSoC Crypto Hardware
Public Encrypt 2048	4.874	**0.552**
Private Decrypt 2048	89.25	**12.846**
Public Encrypt 4096	18.519	**1.95**
Private Decrypt 4096	619.47	**95.9**

wolfSSL on Linux

wolfSSL version 3.12.0 was run under Linux 4.14 as available in Xilinx release 2018.3[Ref 5] for the Zynq UltraScale+ MPSoC. This was used to benchmark the software-only solution, Arm v8 cryptographic extensions, and the Zynq UltraScale+ MPSoC's embedded cryptographic core performance of the AES-GCM 256-bit, SHA3/384, RSA-2048, and RSA-4096 algorithms. Due to memory management and execution privilege differences, applications running under Linux are unlike RTOS applications, in that the same modifications made to the FreeRTOS port were not applied to the Linux port. However, the same data set used in the FreeRTOS benchmarking was used in Linux.

Due to Linux, the expectation is that user space applications run at the lowest-exception level (EL0) on the Armv8 architecture. The Zynq UltraScale+ MPSoC's cryptographic extensions are only accessible from user space applications via a secure monitor call (SMC). This accesses the platform management unit (PMU), which then accesses the cryptographic operation in the CSU using the XilSecure library[Ref 4]. When the cryptographic operation is complete, the Linux application continues to run after returning from the CSU, returning then from the PMU call, and subsequently from the SMC. The high overhead of this scheme is reflected in the Linux benchmarking results.

Results

Table 5 shows the results of AES-GCM 256-bit encryption running the wolfSSL benchmarks under Linux. This table highlights in **bold** the fastest performing cryptographic extension and shows that the Arm v8 cryptographic extensions outperform the Zynq UltraScale+ MPSoC's embedded cryptographic core for all data block sizes. This is because the Arm v8 cryptographic extensions can be directly accessed in user space and do not have to traverse a large software stack to reach the CSU.

Table 5: **Linux Throughput, AES-GCM Encryption**

Block Size	Linux 4.14 Operation Throughput: AES-GCM 256-bit Encryption (MB/s)		
	Software Only	Arm v8 Crypto Extensions	Zynq UltraScale+ MPSoC Crypto Hardware
16	4.426	**54.309**	0.151
528	8.971	**300.263**	4.68
1,024	9.105	**322.934**	8.66
4,112	9.206	**344.002**	27.769
7,696	9.235	**347.552**	42.572
15,888	9.23	**345.605**	59.948

Table 6 shows the results of AES-GCM 256-bit decryption running the wolfSSL benchmarks under Linux. This table highlights in **bold** the fastest performing cryptographic extension and shows that the Arm v8 cryptographic extensions outperform the Zynq UltraScale+ MPSoC's embedded cryptographic core for all data block sizes, due to the same reasons stated for encryption.

Table 6: **Linux Throughput, AES-GCM Decryption**

Block Size	Linux 4.14 Operation Throughput: AES-GCM 256-bit Decryption (MB/s)		
	Software Only	Arm v8 Crypto Extensions	Zynq UltraScale+ MPSoC Crypto Hardware
16	4.397	**34.835**	0.146
528	8.955	**179.661**	4.535
1,024	9.1	**192.258**	8.52
4,112	9.22	**203.652**	27.083
7,696	9.231	**205.579**	41.784
15,888	9.213	**204.456**	59.311

In Table 7, the performance of SHA3/384 reveals that the software-only solution outperforms the Zynq UltraScale+ MPSoC's embedded cryptographic core for data block sizes less than or equal to 1,024 bytes, and the MPSoC crypto core outperforms the software-only solution for data block sizes greater than or equal to 4,112 bytes.

Table 7: **Linux Throughput, SHA3/384**

Block Size	Linux Operation Throughput: SHA3/384 (MB/s)	
	Software Only	Zynq UltraScale+ MPSoC Crypto Hardware
16	**41.58**	0.308
528	**59.632**	9.822
1,024	**60.797**	18.602
4,112	62.025	**66.507**
7,696	62.357	**112.26**
15,888	62.291	**179.718**

Table 8 shows that the Zynq UltraScale+ MPSoC's embedded cryptographic core outperforms all software-only RSA solutions, even with the high Linux calling overhead.

Table 8: **Linux Average RSA Operation Time**

RSA Operation	Linux Avg. RSA Operation Time: wolfSSL v3.12.0 (ms)	
	Software Only	Zynq UltraScale+ MPSoC Crypto Hardware
Public Encrypt 2048	4.424	**1.342**
Private Decrypt 2048	83.512	**25.71**
Public Encrypt 4096	16.736	**4.152**
Private Decrypt 4096	569.89	**191.778**

Conclusion

With the exception only of the SHA3/384 Linux result, the cryptographic acceleration provided by the Arm v8 cryptographic extensions and by the Zynq UltraScale+ MPSoC's embedded cryptographic core outperforms a software-only solution.

Table 9 highlights the percentage of maximum performance improvement using the cryptographic accelerators over a software-only solution. In Table 9, all maximum percentage of performance improvements occurred in the wolfSSL FreeRTOS benchmark results using the Zynq UltraScale+ MPSoC's embedded cryptographic core with the largest block size for the AES and SHA algorithms, and for all key sizes and all operations for the RSA algorithm.

Table 9: **Performance Improvement: Cryptographic Algorithms Over Software-Only Solutions**

Algorithm	Performance Improvement
AES-GCM 256-bit Encryption	10,800.39%
AES-GCM 256-bit Decryption	4,720.07%
SHA3/384	1,037.98%
RSA 2048 Public Encryption	782.97%
RSA 2048 Private Decryption	594.77%
RSA 4096 Public Encryption	849.69%
RSA 4096 Private Decryption	545.95%

The slight performance improvement of the software-only solutions between the FreeRTOS and Linux data is due to the fact that the software-only solutions used different versions of compilers. The FreeRTOS benchmarking used GCC version 6.2.1, while the Linux benchmarking used GCC version 7.3.1. To confirm this difference in compilers, the software-only version of wolfSSL for Linux was recompiled using GCC version 6.2.1 and the AES-GCM 256-bit algorithms were retested. The results showed that AES-GCM 256-bit software-only encryption performance decreased to 7.023 MB/s, while decryption performance decreased to 7.022 MB/s. Even though this white paper focuses on using cryptographic accelerators, the difference in compiler version performance shows that *using the latest tools is extremely important* to maximize software performance.

Figure 1 compares all the benchmarking results from this white paper of the AES-GCM 256-bit encryption algorithm run under the different benchmarking scenarios. Figure 1 clearly shows that the software-only solutions have dramatically slower throughput compared to both accelerated cryptographic solutions. When running wolfSSL benchmarking under FreeRTOS, the Zynq UltraScale+ MPSoC's hardened cryptographic core outperforms the Arm v8 cryptographic extensions for block sizes greater than 2,000 bytes. The FreeRTOS software-only solution overlaps with the Linux software-only solution and is not legible on the chart. Due to the high overhead of calling the MPSoC's AES-GCM 256-bit encryption engine from Linux, using the Arm v8 cryptographic extensions is always a better option to maximize performance.

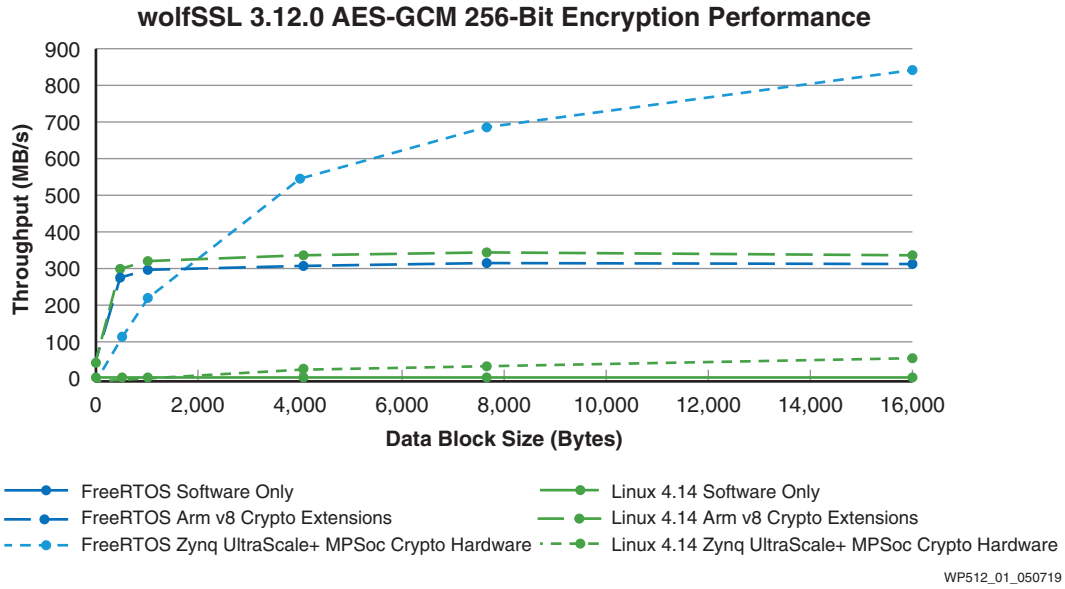

Figure 1: **Comparison Summary of All AES-GCM 256-Bit Encryption Results**

Figure 2 compares all the performance results of the AES-GCM 256-bit decryption algorithm run under the different benchmarking scenarios. Figure 2 does show results similar to those displayed in Figure 1; however, the performance of the Zynq UltraScale+ MPSoC's embedded cryptographic engine run under FreeRTOS outperforms the Arm v8 cryptographic extensions for block sizes greater than 4,000 bytes. Again, the FreeRTOS software-only solution overlaps with the Linux software-only solution and is not legible on the chart.

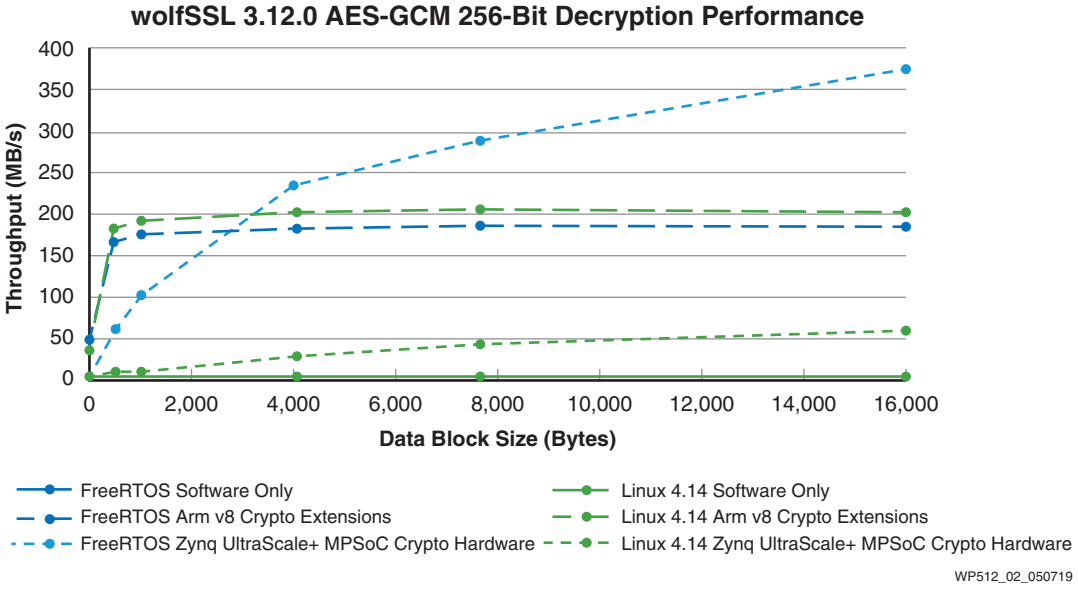

Figure 2: **Comparison Summary of All AES-GCM 256-Bit Decryption Results**

Figure 3 shows the comparison results of running the SHA3/384 algorithm under the different benchmarking scenarios. Like the results for AES-GCM 256-bit decryption, the performance of the Zynq UltraScale+ MPSoC's embedded cryptographic engine when run under FreeRTOS clearly outperforms all the other solutions except when using a block size of 16 bytes. Even Linux, with its

high calling overhead, shows performance improvement over a software-only solution when using the embedded cryptographic accelerators for block sizes greater than 4,000 bytes.

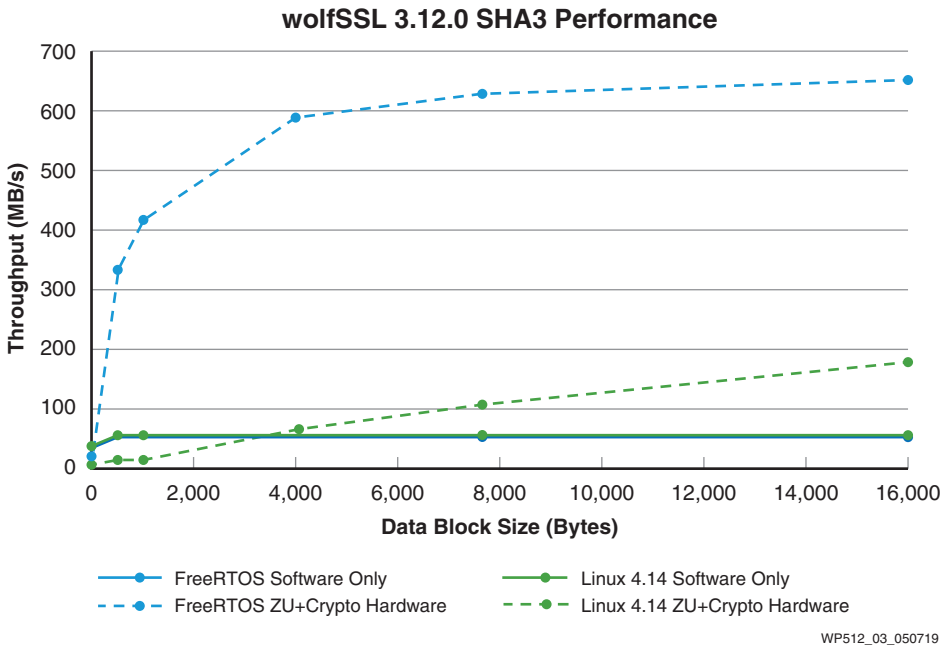

Figure 3: **Comparison Summary of All SHA3/384 Results**

Figure 4, 5, 6, and 7 show the collective comparison of running all RSA operations under the various benchmarking scenarios. These graphs consistently indicate that *the use of the Zynq UltraScale+ MPSoC's embedded cryptographic core always outperforms a software-only solution*. Additionally, when accelerated performance is measured running RSA algorithms under FreeRTOS versus Linux, FreeRTOS is always at least twice as fast due to the CSU being directly accessible in FreeRTOS.

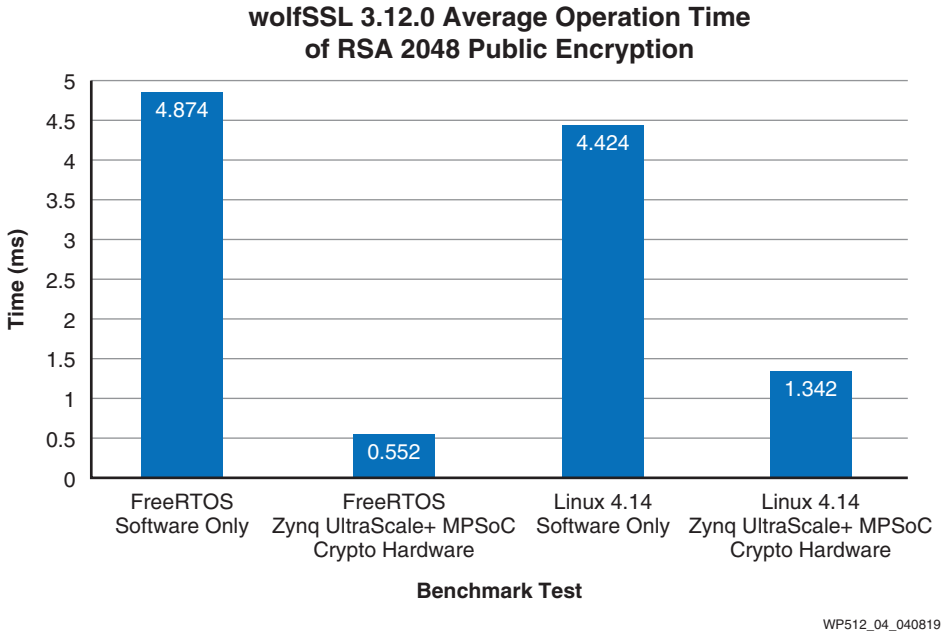

Figure 4: **Summary of All RSA-2048 Public Encryption Benchmark Results**

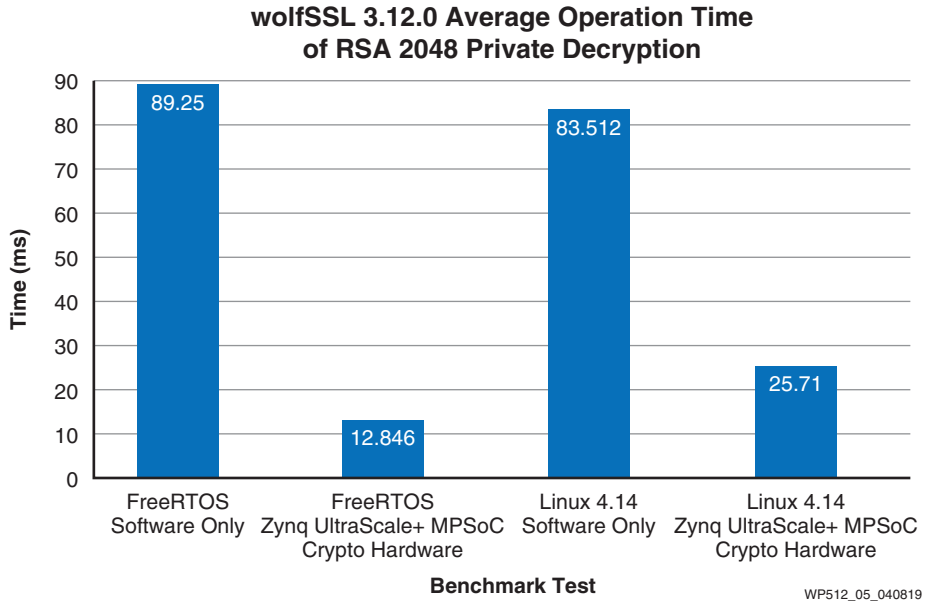

Figure 5: **Summary of All RSA-2048 Public Decryption Benchmark Results**

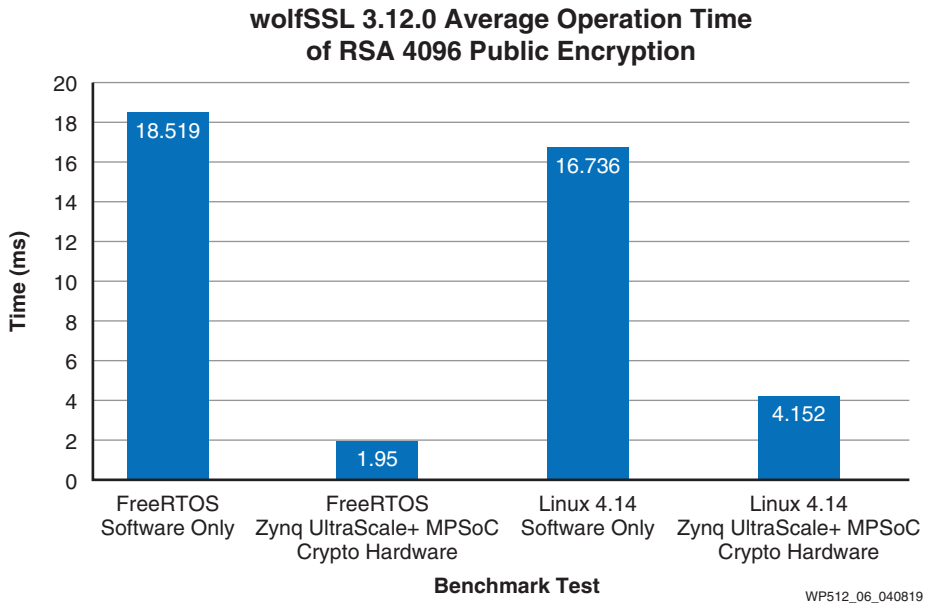

Figure 6: **Summary of All RSA-4096 Public Encryption Benchmark Results**

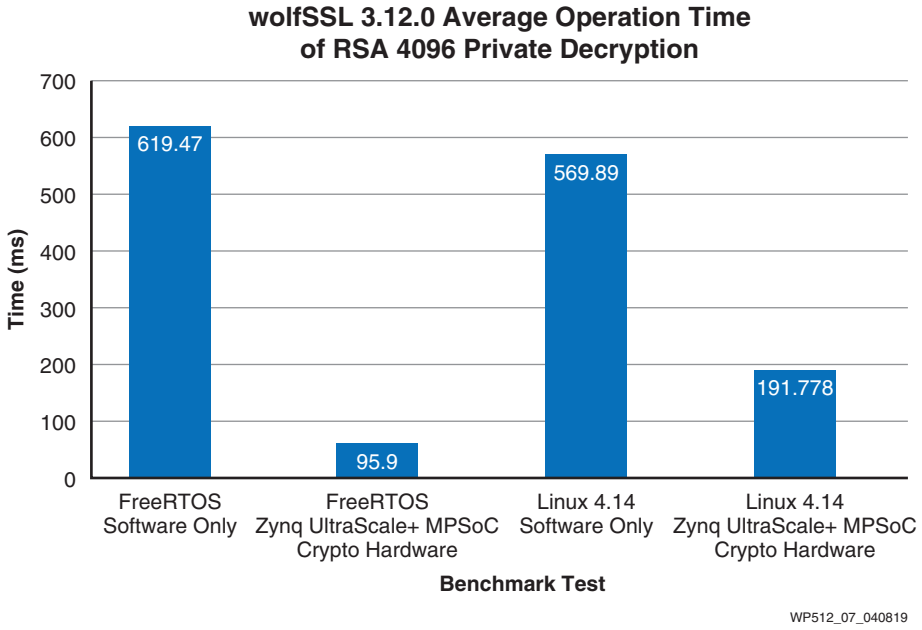

Figure 7: **Summary of All RSA-4096 Public Decryption Benchmark Results**

System architects who are tasked to define cryptographic solutions should evaluate:

- *How the algorithm is run*—utilizing Arm v8 cryptographic extensions, or using the Zynq UltraScale+ MPSoC's embedded cryptographic core
- *The size of each data block*
- *Where the algorithm is running*—on a simple RTOS, or in Linux user space
- *Out of which memory space the software is running*—out of DDR memory, or out of on-chip memory (OCM). While this paper did not test running the cryptographic algorithms out of DDR versus OCM, this factor should be considered.

As shown in this white paper, these variables can significantly affect the performance of the Zynq UltraScale+ MPSoC's accelerated cryptographic performance. The research for this white paper covered only a small subset of possible benchmarking scenarios compared to the number of possible solution combinations. End users should ultimately benchmark their own design to make sure performance requirements are being met.

These results confirm that using the cryptographic accelerators on the Zynq UltraScale+ MPSoC can vastly improve cryptographic performance and will benefit any market and any application that utilizes the AES-GCM-256-bit algorithm, the SHA3/384 algorithm, the RSA-2048 algorithm, and/or the RSA-4096 algorithm. Secondly, these algorithms cover all the foundations for a secure system and covers the aspects of confidentiality, integrity, and authentication. Lastly, not only is a performance increase achieved on the Zynq UltraScale+ MPSoC by using the cryptographic accelerators but doing so will reduce the resource load on the Arm Cortex-A53s, the Arm Cortex-R5s, and the programmable logic. In this way, Zynq UltraScale+ MPSoC applications have a lot more system resources available to dedicate to the application's needs instead of devoting resources to cryptographic needs.

References

1. Xilinx User Guide UG1085, *Zynq UltraScale+ Device Technical Reference Manual*
2. Arm Technical Reference Manual, *Cortex®-A53 MPCore Processor Cryptography Extension*, Revision r0p4
3. wolfSSL website product landing page
4. Xilinx User Guide UG1137, *Zynq UltraScale+ MPSoC Software Developer Guide*, Appendix I: XilSecure Library v3.1
5. Xilinx User Guide UG1144, *Linux Tools Documentation Reference Guide*

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
05/21/2019	1.0	Initial Xilinx release.

Disclaimer

Automotive Applications Disclaimer