Online-Journals.org

iJIM International Journal of Interactive Mobile Technologies

iJIM | eISSN: 1865-7923 | Vol. 18 No. 13 (2024) | OPEN ACCESS

https://doi.org/10.3991/ijim.v18i13.49153

PAPER

Wearable Processors Architecture: A Comprehensive Analysis of 64-bit ARM Processors

Ameen Shaheen[1], Wael Alzyadat[1], Ala'a Al-Shaikh[2](✉), Aysh Alhroob[1]

[1]Faculty of Science and Information Technology, Al-Zaytoonah University, Amman, Jordan

[2]Faculty of Information Technology, Zarqa University, Zarqa, Jordan

ashaikh@zu.edu.jo

ABSTRACT

Wearable devices are playing an important role in our daily lives. Nowadays, wearable devices have transformative implications for health, technology, connectivity, human-computer interaction, and data analytics. Their importance lies in their ability to enhance various aspects of life and contribute to the ongoing evolution of digital landscapes. At the heart of smartwatch design, the processor takes center stage, driving the majority of advancements in smartwatch technology. This paper presents an experimental comparative study of ARM 64-bit processors, analyzing their performance and impact on power consumption, CPU usage, and battery temperature. We evaluate the characteristics of four smartwatch processors: Snapdragon W5+, Snapdragon Wear4100, Exynos W920, and Exynos W930. All of those smartwatches are equipped with ARM 64-bit processors. Our results indicate that none of the four selected smartwatches excelled in all aspects; each exhibits superiority over the others in specific features while being surpassed by others in different attributes.

KEYWORDS

wearable processors, ARM 64-bit, smartwatch

1 INTRODUCTION

The widespread adoption of wearable devices in our daily lives has led to the development of advanced processing technologies to meet the increasing demands across a wide range of applications [1]. As these devices continue to permeate various aspects of our lives, optimizing the delicate balance between performance, power consumption, and thermal management emerges as a paramount concern [2, 3, 4]. This research aims to conduct a comprehensive investigation, providing a comparative study that analyzes the intricate interactions among wearable processors. Special emphasis is placed on enhancing power consumption, central processing unit (CPU), battery temperature, and overall performance.

In the evolving realm of wearable technology, the challenge of enhancing performance without compromising the device's power efficiency or thermal equilibrium

Shaheen, A., Alzyadat, W., Al-Shaikh, A., Alhroob, A. (2024). Wearable Processors Architecture: A Comprehensive Analysis of 64-bit ARM Processors. *International Journal of Interactive Mobile Technologies (iJIM)*, 18(13), pp. 82–95. https://doi.org/10.3991/ijim.v18i13.49153

Article submitted 2024-02-16. Revision uploaded 2024-03-29. Final acceptance 2024-04-06.

is a significant imperative [5]. This study seeks to explain the nuanced relationships in the design and implementation of wearable processors, shedding light on the strategies employed to enhance computational capabilities while preserving battery life and addressing thermal challenges.

Our investigation involves an accurate examination of various wearable processors prevalent in contemporary devices, encompassing both established and emerging architectures [6]. Through a systematic evaluation, we aim to measure the efficiency of these processors in terms of key performance indicators such as computational efficiency, multitasking capabilities, and responsiveness [7, 8]. At the same time, we delve into the intricate dynamics of power consumption, examining energy efficiency and resource utilization during typical usage scenarios. The impact of enhanced performance on the CPU and battery temperature is also a central point, necessitating an in-depth analysis of the thermal management strategies employed by these processors [9].

By undertaking this comparative study, our objective is to provide valuable insights into the delicate balance that wearable processors must achieve between optimal performance, energy efficiency, and thermal stability. The outcomes of this research aim to inform the ongoing discourse on wearable technology, provide practical considerations for developers, and guide the trajectory of future advancements in this dynamic field.

2 RELATED WORKS

The Von Neumann architecture is the foundational and all-encompassing computer architecture in the realm of computational science. Essentially, the Von Neumann computing system comprises three primary components: the CPU, memory, and input/output (I/O) interfaces [10, 11]. A visual representation of the components of the Von Neumann architecture is illustrated in Figure 1. An inherent limitation of the Von Neumann architecture lies in the memory wall dilemma, characterized by the noticeable difference between CPU processing speed and memory access latency, resulting in a bottleneck within the system [12, 13]. One potential solution for addressing the memory wall dilemma is to integrate additional cache levels into the system.

The Harvard architecture represents an alternative computer architecture that offers significant refinements compared to the Von Neumann architecture. It finds frequent application in embedded systems, particularly those requiring low power consumption and high reliability standards [12]. The distinguishing feature of the Harvard architecture lies in its utilization of two separate memory units: one dedicated to program storage and the other designated for data handling [13]. The primary differentiation between these two architectural paradigms is visually exemplified in Figure 2.

The Harvard architecture incorporates separate pathways or buses for each of its memory units: one dedicated to data memory and another designated for instruction memory [14]. This deliberate separation of pathways addresses a longstanding bottleneck issue inherent in the Von Neumann architecture, where a single pathway serves both data and instructions. As a result of this architectural distinction, overall system performance is markedly improved [15].

Pipelining facilitates the concurrent execution of multiple instructions by orchestrating the overlap of instruction execution [16]. Instruction-level parallelism (ILP) involves overlapping instruction execution, allowing instructions to be executed

concurrently and speeding up the overall execution process [17, 18]. Conventionally, pipelines consist of five distinct stages: (1) fetch, (2) decode, (3) execute, (4) memory, and (5) write back [19]. The number of stages in a pipeline is commonly referred to as the pipeline's depth [20]. Deeper pipelines, characterized by an increased number of stages, enable the implementation of each stage with simplified circuitry, resulting in a higher processor clock speed [21].

A compiler, a category of system software, assumes the responsibility of translating programs written in various programming languages into machine code. Consequently, programs are executed in either:

- In an in-order execution model, instructions are dispatched and executed in the sequence in which they are presented within the program. One drawback associated with this model is the presence of interdependencies among instructions. This means that if one instruction encounters a stall, it will lead to the stalling of all instructions dependent on the stalled one. Consequently, program execution remains suspended until the stalled instruction can proceed [22, 23].
- In contrast, the out-of-order execution model, also known as dynamic execution or dynamic scheduling [24], represents a performance enhancement technique or execution paradigm employed to improve overall system performance [25].

The process of instruction fetching follows a sequential order. Similarly, instructions conclude their execution in the same order in which they were initiated. However, it is essential to note that the execution of instructions may not strictly follow the original order, thereby giving rise to the term "out-of-order execution" [26].

It is worth emphasizing that the use of dynamic scheduling introduces additional complexities compared to the previously used static scheduling, which involves scheduling at compile-time. This increase in scheduling complexity, in turn, has a significant impact on power consumption [27].

Nonetheless, out-of-order execution offers a significant advantage in terms of faster processing, although it requires a more significant allocation of core logic, an improved CPU architecture, and an expanded register set to support this approach. In their study, [28] conducted a comprehensive performance analysis of 64-bit ARM processors deployed within a cluster of computers designed for high-performance computing (HPC). This ARM-based cluster's performance was compared with that of an x86 Intel Ivy Bridge system. The energy-delay product (EDP) is calculated as the product of energy consumption and system performance. The findings of this analysis indicate a positive correlation between the number of processor cores and the efficiency achieved.

The study examined single-board computers in the context of their utilization within educational service robots. The research encompassed an exploration of various processors, including the ARM-based processor, the Raspberry Pi 3B+. The investigation findings revealed that, although the Raspberry Pi processor has the highest power consumption levels, it continues to be the primary choice in the field of robotics.

This preference is attributed to its extensive community, robust support infrastructure, and the wide array of available software packages [29]. In [30], the authors investigated the impact of computational tasks on energy consumption, specifically assessing the utilization of both CPUs and graphics processing units (GPUs) within ARM MPSoC platforms. Their work involved a comprehensive comparative analysis of power efficiency between CPUs and GPUs integrated into system-on-chip (SoC) architectures. In a similar vein, Qasaimeh and colleagues [31] undertook a performance

evaluation of three distinct hardware accelerators for embedded vision applications. Furthermore, [32] delved into the assessment of ARM processors' performance in the context of deep learning, with particular attention directed toward ARM Cortex-A57 and Cortex-A78AE CPUs, among other processor variants. In a different approach, the research conducted by [33] focused on developing a tool designed to facilitate the estimation of performance and energy consumption for applications executed on embedded devices. They introduced a framework based on static analysis methodologies.

In our study, we focus on the experimental results of ARM 64-bit processors in terms of performance and their impact on power consumption, CPU, and battery temperature.

3 ARM ARCHITECTURE

ARM microprocessors have many generations.

1. The 32-bit processors consist of six main processor families. (ARM10E, ARM9TDMI, ARM7EJ, ARM7TDMI, ARM9E, and ARM Cortex).
2. The 64-bit version is exemplified by the ARMv8-A processor.

In scenarios where code density takes precedence and for high-volume applications facing memory limitations [34], ARM introduced the Thumb architecture. Thumb represents a 16-bit architecture, comprising an instruction set that is a subset of the 32-bit instruction set. From a statistical standpoint, code written in the thumb instruction set is approximately 65% smaller than code composed using the standard 32-bit instruction set. Moreover, when executed from 16-bit memory, the performance of such code demonstrates a 160% increase [35]. The thumb instruction set is well-suited for scenarios involving memory constraints. Additionally, its increased code density makes it optimal for systems that require low power consumption. Certain ARM processors adhere to the Von Neumann architecture, while others are designed based on the Harvard architecture. Nonetheless, variations exist among processors within the same family. In the ARM7TDMI, for instance, the pipeline depth comprised three stages, and the architecture followed the Von Neumann computer architecture [35]. As an enhancement, the ARM9TDMI adopted a more extensive pipeline with five stages and transitioned to the Harvard architecture, representing an improvement over the conventional Von Neumann architecture [36]. Nevertheless, both of them are 32-bit processors.

ARMv7 outlines three architectural profiles.

- Application profile (A-Profile): It supports the virtual memory system architecture (VMSA) using a memory management unit (MMU). This profile, also known as VMSAv7 [21], is designed for high-performance systems capable of running operating systems [37].
- Real-time profile (R-profile): It supports the protected memory system architecture (PMSA) by utilizing of a memory protection unit (MPU). This profile is also known as PMSAv7 [24] and is designed for systems that require deterministic timing and minimal interrupt latency [37].
- Microcontroller profile (M-profile): It constitutes a variation of the PMSAv7, specifically crafted to facilitate low-latency interrupt processing [24].

In real-time systems, a full-fledged MMU for the processor is unnecessary. A PMU would be adequate, as it exclusively handles essential protection operations without

engaging in memory management or address translation [38]. The transition from a 32-bit to a 64-bit architecture is crucial for addressing numerous modern computing challenges. Motivations for transitioning to a higher word width may arise from the need to handle data volumes that exceed the capacity of a specific CPU. In cases where the data width exceeds that of the CPU, impeding its processing capability, migration becomes necessary. Moreover, this could manifest as single-instruction multiple-data (SIMD) processing, a form of parallel processing. Simultaneous fetching of multiple instructions or even data items to the CPU becomes possible when the word width of the data to be transferred is less than the bus width [39]. Therefore, migrating to 64-bit architectures results in a twofold increase in speed and enables the processor to access a larger physical memory, surpassing the 4GB threshold. This, in turn, enhances computational capabilities and achieves higher performance [39]. ARMv8 represents a 64-bit iteration of ARM, featuring backward compatibility for 32-bit ARMv7 programs and applications. The transition involves deeper pipelines with out-of-order, speculative, and superscalar execution [42]. ARM microprocessor designs utilize the big.LITTLE architecture, also known as heterogeneous computing architecture. In this configuration, a microprocessor consists of both high-performance big cores and low-power LITTLE cores [40]. The aim is to develop a multi-core processor capable of achieving two primary objectives: (1) high performance and (2) low power consumption. These objectives align with the dual constraints of wearable system design [41]. The transition between big and LITTLE cores is facilitated through inter-cluster core switching, also known as big.LITTLE switching [42, 43].

Figure 1 shows a schematic diagram illustrating the big.LITTLE architecture developed by ARM.

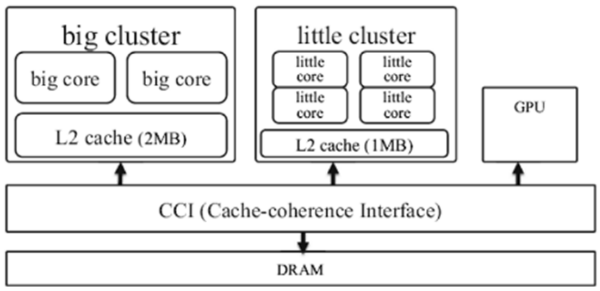

Fig. 1. ARM big.LITTLE (heterogeneous computing) architecture [35]

4 PROCESSOR ARCHITECTURE COMPARISON

In the following, we introduce four types of processors from different vendors. All of them contain several ARM cores. The processors are: (1) Snapdragon W5+, (2) Snapdragon Wear 4100, (3) Exynos W920, and (4) Exynos W930.

4.1 Snapdragon W5+

The Snapdragon W5+ by Qualcomm Technologies is an advanced wearable system-on-chip (SoC) with a quad-core, 64-bit processor operating at up to 1.7 GHz [44]. It is designed for comprehensive energy efficiency, featuring a hybrid architecture with a cutting-edge 4 nm SoC and an integrated 22 nm AON Co-Processor. The platform incorporates Bluetooth 5.3, dedicated low-power islands for Wi-Fi, GNSS,

and audio, as well as energy-efficient states such as deep sleep and hibernate. These innovations result in a significant 30–60% reduction in power consumption during typical usage, leading to an impressive 50% or more extension in battery life. Notably, the Snapdragon W5 Gen 1 platform deliberately omits using the co-processor, relying exclusively on the inherent low-power capabilities of the SoC [45].

4.2 Snapdragon Wear 4100

Snapdragon Wear 4100 is a high-performance wearable SoC developed by Qualcomm Technologies. It features a quad-core, 64-bit processor, each running at a clock speed of 2.0 GHz [46].

The newly implemented platforms employ 12 nm low-power process technology, incorporate dual DSPs for efficient workload partitioning, integrate Qualcomm® Sensor Assisted Positioning for wearables, and utilize Bluetooth 5.0 with the specific objective of achieving a reduction in power consumption exceeding 25% and enhancing the overall battery life of the platform [46].

4.3 Exynos W920

The Exynos W920 is a dual-core 64-bit processor that utilizes a 5 nm process and incorporates ARM Cortex-A55, each with a clock speed of up to 1.18 GHz [47]. Both the Snapdragon W5+ and the Exynos W920 used the always-on-display (AOD) feature.

4.4 Exynos W930

The Exynos W930 is a dual-core 64-bit processor that utilizes the 5 nm manufacturing process and includes two ARM Cortex-A55, each with a clock speed of up to 1.4 GHz [48].

The features of the four processors that were chosen are summarized in Table 1.

Table 1. Processors feature

Brand / Factor	Snapdragon W5+	Snapdragon Wear 4100	Exynos W920	Exynos W930
Cores	4	4	2	2
CPU	4 × cortexA53	4 × cortexA53	2 × cortex-A55	2 × cortex-A55
CPU clock	4 × 1.7 cortex-A53	4 × 2.0 cortex-A53	2 × 1.18 cortex-A55	2 × 1.4 cortex-A55
Architecture	64-bit			
Battery capacity	625 mAH	600 mAH	415 mAH	300 mAH

5 EXPERIMENTAL RESULTS

The experiments are conducted on four smartwatches, namely: (1) Oppo watch 3 equipped with a Snapdragon W5+ processor; (2) Mobvoi Ticwatch E3 with a Snapdragon Wear 4100 processor; (3) Galaxy Watch 5 with an Exynos W920 processor; and (4) Galaxy Watch 6 with an Exynos W930 processor. There are no specific

selection criteria for the smartwatches, except that we aimed to experiment with all four types of processors. The performance tests are configured to run using the Mozilla Kraken benchmark.

5.1 Performance

This performance benchmark, devised by Mozilla, quantifies the speed of various test cases extracted from real-world programs and libraries during the processing phase.

Figure 2 displays the results of running Mozilla Kraken on the four CPUs mentioned earlier.

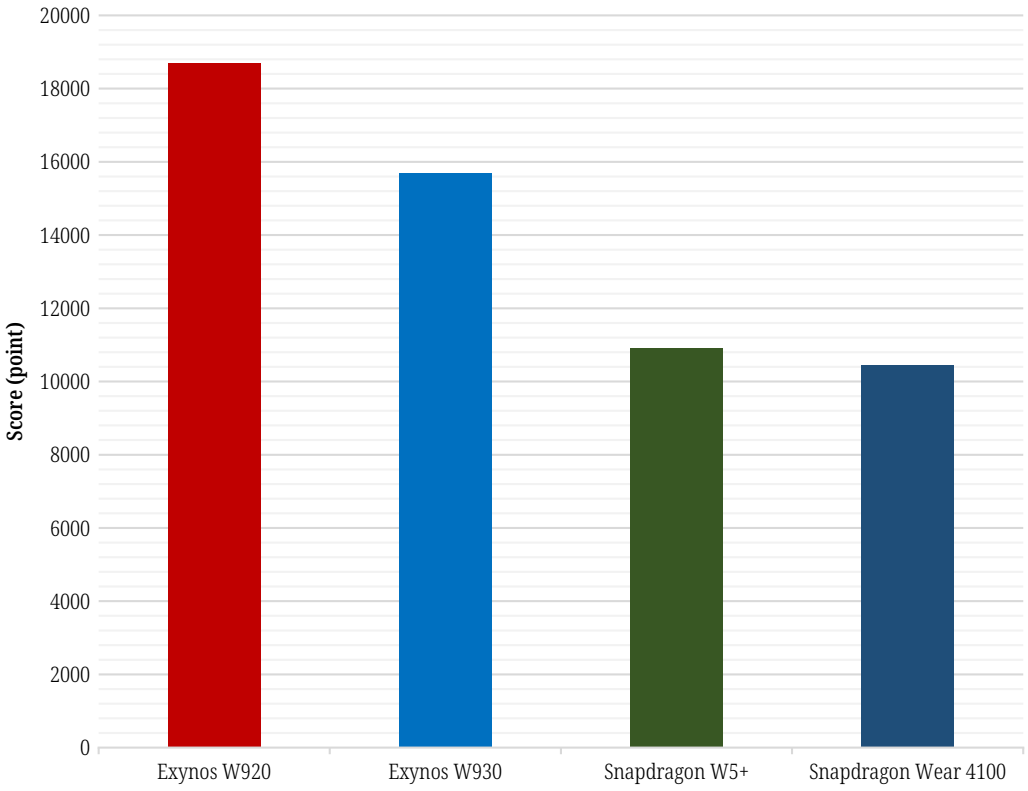

Fig. 2. Mozilla Kraken scores

As shown in Figure 2, it is evident that Exynos consistently achieved superior scores compared to Snapdragon. This discrepancy arises from the enhanced specifications inherent in the Cortex-A55 processors employed by Exynos, as opposed to the Cortex-A53 utilized in the Snapdragon W5+ and Snapdragon Wear 4100.

The efficiency of a processor is directly proportional to the reduction in transistor size; therefore, smaller transistors contribute to improved efficiency. In the case under consideration, the Snapdragon W5+ utilizes a manufacturing process with 4 nm transistors, whereas the Snapdragon Wear 4100 employs transistors with a size of 12 nm. Notably, both the Exynos W920 and Exynos W930 feature a more advanced 5 nm EUV lithography process, introducing a discernible distinction. Despite the Snapdragon W5+ employing a 4 nm process, the Exynos counterparts, with their 5 nm EUV lithography process, demonstrate a significant

reduction in power consumption due to the integration of low-power CPU design technologies.

5.2 CPU and battery temperature

This examination assesses the temperatures of the CPU and battery under heightened CPU load conditions. Figure 3 delineates the CPU temperature in idle and moderate use, while Figure 4 illustrates the battery temperature. It is noteworthy that lower values in both figures correspond to more favorable outcomes.

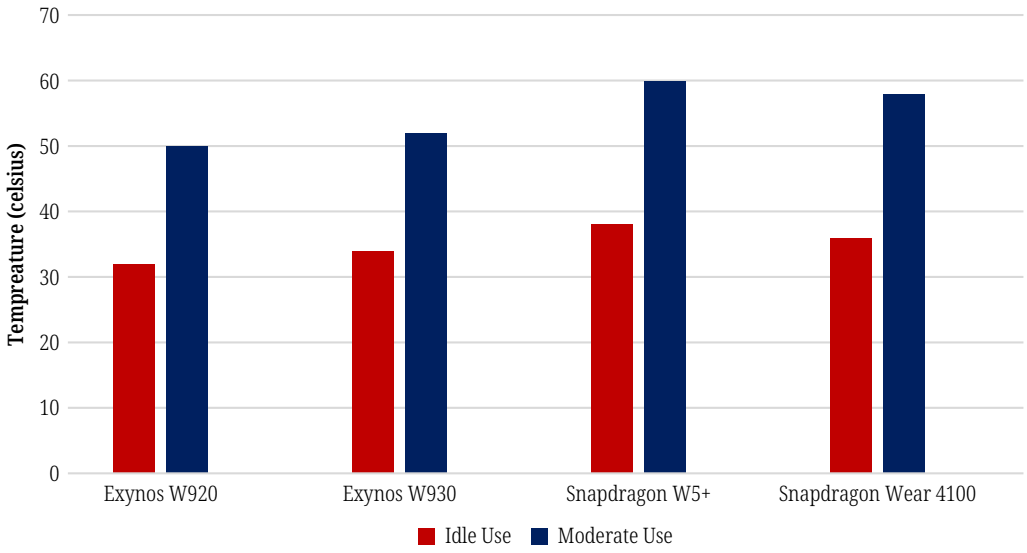

Fig. 3. CPU temperature

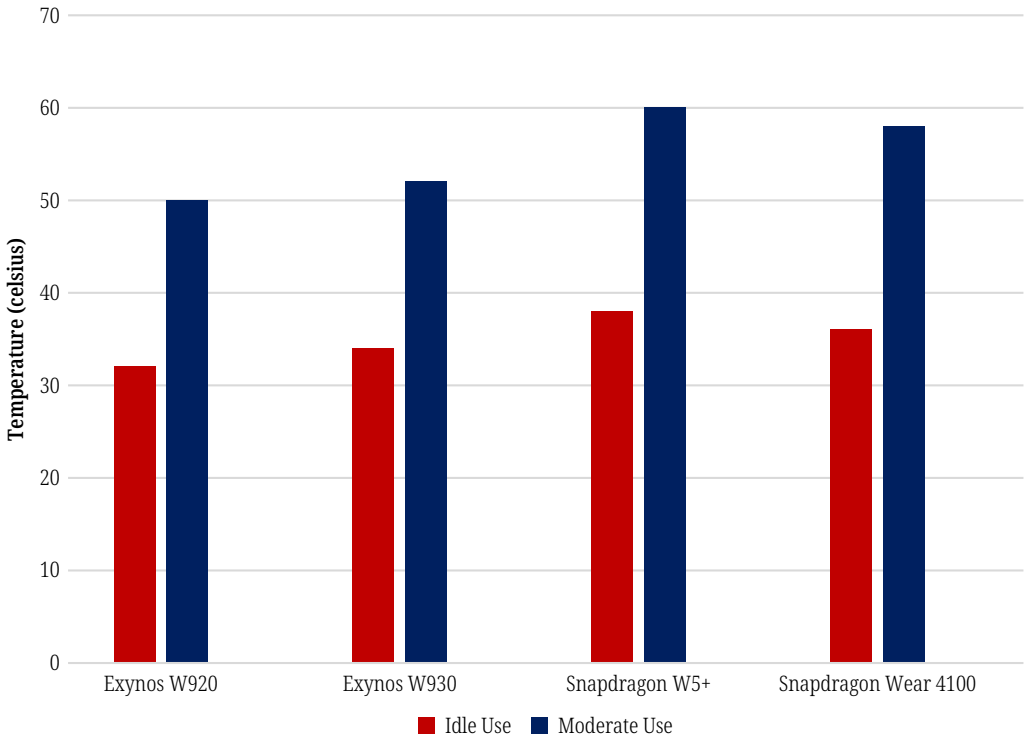

Fig. 4. Battery temperature

Despite the larger battery capacity in both the Snapdragon W5+ and Snapdragon Wear 4100, the Exynos W920 and Exynos W930 exhibit lower CPU and battery temperatures.

This phenomenon can be attributed to the activation of a dedicated low-power display processor, namely the Cortex-M55, in both the Exynos W920 and Exynos W930.

This integration effectively reduces display power consumption under the AOD mode compared to their respective predecessors in the Exynos lineup.

5.3 Power consumption

This test has been conducted to compare power consumption capabilities as the CPU load increases. The values are derived through the measurement of total energy consumption conducted during individual tests, each lasting for two minutes. To ensure the precision and reliability of the acquired data, the test procedure is repeated five times. Subsequently, the obtained results are averaged to consolidate the readings. Figure 5 displays the results for four CPU load values.

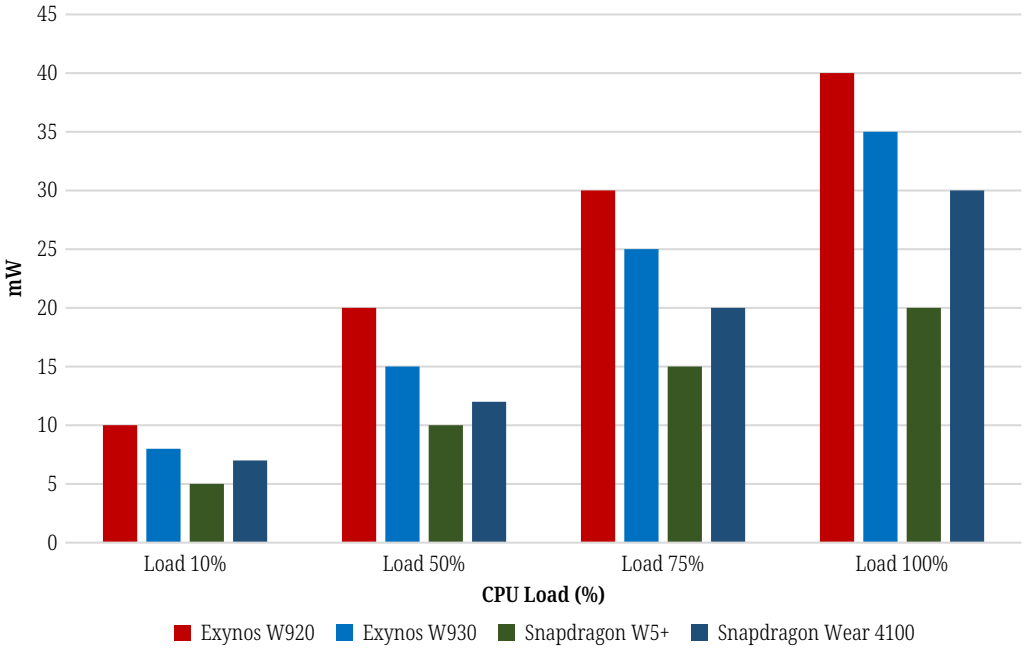

Fig. 5. Power consumption

As illustrated in Figure 5, it is evident that the Snapdragon W5+ exhibits the lowest power consumption, despite having the highest recorded CPU and battery temperatures. Typically, elevated temperatures are associated with increased power consumption; however, the Snapdragon W5+ is engineered to demonstrate remarkably low power consumption across various operational levels. The platform features an enhanced hybrid architecture distinguished by an innovative 4 nm SoC and a highly integrated 22 nm always-on network (AON) co-processor, thereby highlighting the unique attributes that contribute to this observed phenomenon. This distinctly influences the observed outcomes in this context. Snapdragon Wear 4100 comes in second place, Exynos W930 comes in third place, followed by Exynos W920, which comes in last place.

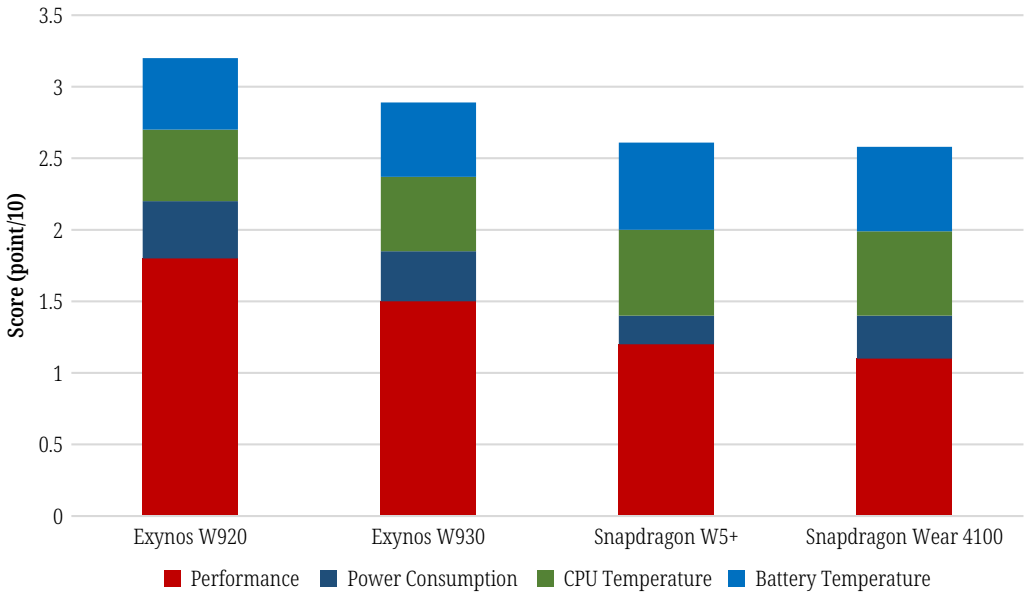

Fig. 6. Exynos W920, Exynos W930, Snapdragon W5+ and Snapdragon Wear 4100 comparison

As a result of the study, we can divide the findings into two categories based on their features. The first category is the high-performance category, which includes both Exynos W920 and Exynos W930. The two processors exhibit slight differences between them because the Exynos W920 utilizes the 5 nm EUV lithography process. The second category is the low-power consumption category, which includes the Snapdragon W5+ and Snapdragon Wear 4100. In conclusion, each SOC has pros and cons. For instance, Figure 6 shows that the Exynos W920 has higher CPU performance than the Snapdragon W5+. On the other hand, Snapdragon achieves lower device power consumption. Concerning the temperature of the CPU and battery, the Exynos W920 runs 10 degrees Celsius cooler than the Snapdragon W5+, which is a significant difference. This temperature advantage can be considered a drawback for the Snapdragon W5+. The Exynos W930 also has higher CPU performance compared to the Snapdragon W5+ and Snapdragon Wear 4100. Similarly, the Exynos W920 offers the highest performance, followed by the Exynos W930, Snapdragon W5+, and the lowest performance is seen in the Snapdragon Wear 4100. The Exynos W920 exhibited the lowest CPU and battery temperatures, followed by the Exynos W930. In contrast, the Snapdragon Wear 4100 and Snapdragon W5+ recorded the highest temperatures.

6 CONCLUSION

ARM processors have combined the benefits of a reduced instruction set computing (RISC) architecture while integrating non-trivial components. The strategic alignment of architecture and the inherent simplicity of ARM cores have positioned them as the predominantly utilized processor cores in the field of mobile computing.

In this paper, we evaluated ARM CPU designs and examined how design development influenced the performance of smartwatches, including CPU and battery temperature and power consumption. We studied the advantages of the 64-bit architecture. We considered how CPU and battery temperature affect performance and power consumption. The result indicates that successive design iterations of

processors in smartwatches have shown significant improvements in performance, attributed to the implementation of assertive scaling techniques.

We believe that smartwatches with the same processors show minimal differences. Consequently, conducting experiments on different devices yields results with no notable differences.

In future research endeavors, scholars may employ machine learning and deep learning techniques to predict power consumption and processor efficiency, considering the architectural characteristics of the processors. Subsequent studies could further explore novel processors, facilitating additional comparisons based on the outcomes of the present investigation.

7 REFERENCES

[1] B. Wang, "Evaluation of sports visualization based on wearable devices," *Int. J. Emerg. Technol. Learn. (iJET)*, vol. 12, no. 12, pp. 119–126, 2017. https://doi.org/10.3991/ijet.v12i12.7962

[2] H. Liu, Z. Wei, W. He, and J. Zhao, "Thermal issues about Li-ion batteries and recent progress in battery thermal management systems: A review," *Energy Conversion and Management*, vol. 150, pp. 304–330, 2017. https://doi.org/10.1016/j.enconman.2017.08.016

[3] W. Alzyadat, A. AlHroob, I. H. Almukahel, M. Muhairat, M. Abdallah, and A. Althunibat, "Big data, classification, clustering and generate rules: An inevitably intertwined for Prediction," in *2021 International Conference on Information Technology (ICIT)*, Amman, Jordan, 2021, pp. 149–155. https://doi.org/10.1109/ICIT52682.2021.9491733

[4] A. T. Imam, A. Alhroob, and W. Jumah, "SVM machine learning classifier to automate the extraction of SRS elements," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 3, 2021. https://doi.org/10.14569/IJACSA.2021.0120322

[5] M. S. Tasjid and A. A. Marouf, "Leveraging smartphone sensors for detecting abnormal gait for smart wearable mobile technologies," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 24, pp. 167–175, 2021. https://doi.org/10.3991/ijim.v15i24.25891

[6] M. Yoshimoto and S. Izumi, "Recent progress of biomedical processor SoC for wearable healthcare application: A review," *IEICE Transactions on Electronics*, vol. E102.C, no. 4, pp. 245–259, 2019. https://doi.org/10.1587/transele.2018CDI0001

[7] M. R. Nakhkash, T. N. Gia, I. Azimi, A. Anzanpour, A. M. Rahmani, and P. Liljeberg, "Analysis of performance and energy consumption of wearable devices and mobile gateways in IoT applications," in *COINS '19: Proceedings of the International Conference on Omni-Layer Intelligent Systems*, pp. 68–73, 2019. https://doi.org/10.1145/3312614.3312632

[8] W. Alzyadat and A. AlHroob, "Development planning in the big data era: Design references architecture," *International Journal of Recent Technology and Engineering*, vol. 8, no. 1c2, 2019.

[9] M. Injadat, "Optimized ensemble model towards secured industrial IoT devices," *arXiv*, no. 2401.05509, 2024. https://doi.org/10.48550/arXiv.2401.05509

[10] M. Keley, A. Khademzadeh, and M. Hosseinzadeh, "Efficient mapping algorithm on mesh-based NoCs in terms of cellular learning automata," *The International Arab Journal of Information Technology*, vol. 16, no. 2, pp. 312–322, 2019.

[11] B. Hillier, "Specifically architectural theory: A partial account of the ascent from building as cultural transmission to architecture as theoretical concretion," *Harvard Architecture Review*, vol. 9, pp. 8–27, 1993.

[12] S. Wang, X. Jiang, F. Bai, W. Xiao, X. Long, Q. Ren, and Y. Kang, "A true process-heterogeneous stacked embedded DRAM structure based on wafer-level hybrid bonding," *Electronics*, vol. 12, no. 5, p. 1077, 2023. https://doi.org/10.3390/electronics12051077

[13] G. Samara, M. Aljaidi, E. A. Daoud, M. Y. Al-Safarini, G. M. Alsayyed, and S. Almatarneh, "Message broadcasting algorithm implementation using mobile long-term evolution and narrow band internet of things over Intelligent Transportation System (ITS)," in *2022 International Arab Conference on Information Technology (ACIT)*, Abu Dhabi, United Arab Emirates, 2022, pp. 1–7. https://doi.org/10.1109/ACIT57182.2022.9994179

[14] S. Misra, A. A. Alfa, S. Olamide Adewale, M. A. Akogbe, and M. O. Olaniyi, "A two-way loop algorithm for exploiting instruction-level parallelism in memory system," *Lecture Notes in Computer Science*, Springer, Cham, vol. 8583, pp. 255–264, 2014. https://doi.org/10.1007/978-3-319-09156-3_19

[15] A. Shaheen, R. Al-Sayyed, and A. Sleit, "Improving visual analyses and communications of ontology by dynamic tree (case study: computer system)," *International Journal of Advanced and Applied Sciences*, vol. 4, no. 5, pp. 62–66, 2017. https://doi.org/10.21833/ijaas.2017.05.011

[16] G. K. Dasebenezer and B. Joselin, "TSO clustered protocol to extend lifetime of IoT based mobile wireless sensor networks," *The International Arab Journal of Information Technology*, vol. 20, no. 4, 2023. https://doi.org/10.34028/iajit/20/4/1

[17] A. Thanganadar and V. Raman, "Integrated shared random key agreement protocol for wireless sensor network," *The International Arab Journal of Information Technology (IAJIT)*, vol. 21, no. 2, pp. 201–210, 2024. https://doi.org/10.34028/iajit/21/2/3

[18] H. Takci and F. Nusrat, "Highly accurate spam detection with the help of feature selection and data transformation," *The International Arab Journal of Information Technology (IAJIT)*, vol. 20, no. 1, pp. 29–37, 2023. https://doi.org/10.34028/iajit/20/1/4

[19] J. L. Hennessy and D. A. Patterson, "Computer architecture: A quantitative approach," Cambridge, Ma: Morgan Kaufmann, 2019.

[20] J. P. Shen, "Modern processor design: Fundamentals of superscalar processors," Long Grove: Waveland Press, 2013.

[21] A. Althunibat, M. Abdallah, M. Amin Almaiah, N. Alabwaini, and T. Ahmad Alrawashdeh, "An Acceptance Model of using Mobile-Government Services (AMGS)," *Computer Modeling in Engineering & Sciences*, vol. 131, no. 2, pp. 865–880, 2022. https://doi.org/10.32604/cmes.2022.019075

[22] W. Alzyadat, A. Shaheen, A. Al-Shaikh, A. Alhroob, and Z. Al-Khasawneh, "A proposed model for enhancing e-bank transactions: An experimental comparative study," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 34, no. 2, pp. 1268–1279, 2024. https://doi.org/10.11591/ijeecs.v34.i2.pp1268-1279

[23] R. D. Barnes, S. J. Patel, E. M. Nystrom, N. Navarro, J. W. Sias, and W. W. Hwu, "Beating in-order stalls with "flea-flicker" two-pass pipelining," in *Proceedings 36th Annual IEEE/ACM International Symposium on Microarchitecture, 2003, MICRO-36*, San Diego, CA, USA, 2003, pp. 387–398. https://doi.org/10.1109/MICRO.2003.1253243

[24] ARM, "ARM® Architecture Reference Manual: ARMv7-A and ARMv7-R edition," *ARM*, 2012.

[25] A. Abusukhon, B. Hawashin, and M. Lafi, "An efficient system for reducing the power consumption in offices using the internet of things," *International Journal of Advances in Soft Computing & Its Applications*, vol. 13, no. 1, 2021.

[26] C. Tan, A. Kulkarni, V. Venkataramani, M. Karunaratne, T. Mitra, and L.-S. Peh, "LOCUS: Low-power customizable many-core architecture for wearables," in *2016 International Conference on Compliers, Architectures, and Sythesis of Embedded Systems (CASES'16)*, Pittsburgh, PA, USA, 2016, pp. 1–10. https://doi.org/10.1145/2968455.2968506

[27] A. Al-Shaikh, A. Shaheen, M. Rasmi Al-Mousa, K. Alqawasmi, A. S. Al Sherideh, and H. Khattab, "A comparative study on the performance of 64-bit ARM processors," *Int. J. Interact. Mob. Technol. (iJIM)*, vol. 17, no. 13, pp. 94–113, 2023. https://doi.org/10.3991/ijim.v17i13.39395

[28] C. T. Rueden *et al.*, "ImageJ2: ImageJ for the next generation of scientific image data," *BMC Bioinformatics*, vol. 18, 2017. https://doi.org/10.1186/s12859-017-1934-z

[29] G. Taffoni, S. Bertocco, I. Coretti, D. Goz, A. Ragagnin, and L. Tornatore, "Low power high performance computing on arm system-on-chip in astrophysics," *Advances in Intelligent Systems and Computing*, pp. 427–446, 2019. https://doi.org/10.1007/978-3-030-32520-6_33

[30] M. Qasaimeh, K. Denolf, J. Lo, K. Vissers, J. Zambreno, and P. H. Jones, "Comparing energy efficiency of CPU, GPU and FPGA implementations for vision kernels," in *2019 IEEE International Conference on Embedded Software and Systems (ICESS)*, Las Vegas, NV, USA, 2019. https://doi.org/10.1109/ICESS.2019.8782524

[31] M. F. Dolz, S. Barrachina, H. Martínez, A. Castelló, A. Maciá, G. Fabregat, and A. E. Tomás, "Performance–energy trade-offs of deep learning convolution algorithms on ARM processors," *The Journal of Supercomputing*, vol. 79, pp. 9819–9836, 2023. https://doi.org/10.1007/s11227-023-05050-4

[32] C. Marantos, K. Salapas, L. Papadopoulos, and D. Soudris, "A flexible tool for estimating applications performance and energy consumption through static analysis," *SN Computer Science*, vol. 2, 2021. https://doi.org/10.1007/s42979-020-00405-7

[33] S. Eum, H. Kim, H. Kwon, M. Sim, G. Song, and H. Seo, "Parallel implementations of ARIA on ARM processors and graphics processing unit," *Applied Sciences*, vol. 12, no. 23, p. 12246, 2022. https://doi.org/10.3390/app122312246

[34] S. Chheda, A. Curtis, E. Siegmann, and B. Chapman, "Performance study on CPU-based machine learning with PyTorch," in *HPC Asia '23 Workshops: Proceedings of the HPC Asia 2023 Workshops*, 2023, pp. 24–34. https://doi.org/10.1145/3581576.3581615

[35] Atmel ES2, "ARM7TDMI (Thumb) Data Sheet," *Atmel ES2*, 1999.

[36] ARM, "ARM7TDMI Technical Reference Manual," *ARM*, 2001.

[37] ARM, "ARM9TDMI Technical Reference Manual," *ARM*, 2000.

[38] ARM, "Architectures, Processors, and Devices: Development Article," *ARM*, 2009.

[39] S. Furber, ARM System-on-Chip Architecture, 2nd ed., Pearson Education Limited, 2000.

[40] D. Brooks and M. Martonosi, "Dynamically exploiting narrow width operands to improve processor power and performance," in *Proceedings Fifth International Symposium on High-Performance Computer Architecture*, Orlando, FL, USA, 1999, pp. 13–22. https://doi.org/10.1109/HPCA.1999.744314

[41] ARM, "ARM Cortex-A Series Programmer's Guide for ARMv8-A," *ARM*, 2015.

[42] L. Arvai, "Application of smartwatches in elderly care with indoor localization functionality," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 5, pp. 174–186, 2021. https://doi.org/10.3991/ijim.v15i05.20435

[43] ARM, "big.LITTLE technology: The future of mobile," *ARM*, 2013.

[44] A. Saeb Al-Sherideh, R. Ismail, M. R. Al-Mousa, K. Al-Qawasmi, A. Al-Shaikh, and H. Awwad, "Development of a secure model for mobile government applications in Jordan," *Journal of Statistics Applications & Probability*, vol. 13, no. 1, 2024. https://doi.org/10.18576/jsap/130110

[45] Snapdragon W5 Plus Gen 1 Wearable Platform | Qualcomm. Accessed: Apr. 05, 2024. https://www.qualcomm.com/products/mobile/snapdragon/wearables/snapdragon-w5-plus-gen-1-wearable-platform.

[46] Snapdragon Wear 4100+ platform. Accessed: Apr. 05, 2024. https://www.qualcomm.com/products/mobile/snapdragon/wearables/snapdragon-wear-4100-plus-platform.

[47] Exynos W920 | Wearable Processor | Samsung Semiconductor Global. Accessed: Apr. 05, 2024. Samsung Semiconductor Global. https://semiconductor.samsung.com/processor/wearable-processor/exynos-w920/.

[48] Exynos W930 | Wearable Processor | Samsung Semiconductor Global. Accessed: Apr. 05, 2024. Samsung Semiconductor Global. https://semiconductor.samsung.com/processor/wearable-processor/exynos-w930/.

8 AUTHORS

Ameen Shaheen is an Assistant Professor of Computer Science in the Department of Software Engineering at the Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Jordan. He received his Ph.D. in Computer Science from the University of Jordan in 2019. His research interests include software engineering, artificial intelligence, cloud computing, algorithms, parallel computing, and distributed systems (E-mail: a.shaheen@zuj.edu.jo).

Wael Alzyadat is an Associate Professor of software engineering. He works at Al-Zaytoonah University in Jordan. His research area encompasses software analysis, intelligent systems, streaming data, and big data. Moreover, he has authored over 45 published articles and obtained two copyrights (E-mail: wael.alzyadat@zuj.edu.jo).

Ala'a Al-Shaikh is an Assistant Professor of Computer Science in the Department of Cybersecurity at the Faculty of Information Technology, Zarqa University, Jordan. He received his PhD in Computer Science from the University of Jordan in 2020. AlShaikh has served in various academic institutions in Jordan and Saudi Arabia. His greatest achievements, of which he is always proud, is the computerization of the General Associate Degree Examination in Jordan. This examination is the second national test in Jordan that awards students who pass it the associate degree in accordance with the Law of Higher Education of Jordan. His research interests encompass computational intelligence, machine learning, artificial intelligence, parallel computing, distributed systems, cloud computing, algorithms, e-learning, and cybersecurity (E-mail: ashaikh@zu.edu.jo).

Aysh Alhroob works as the Dean of the Faculty of Science and Information Technology and as a full Professor in the Department of Software Engineering at AlZaytoonah University in Jordan. He has completed his PhD at the University of Bradford, UK. He has published numerous articles on various computer science and software engineering topics in international journals and conferences. In addition, he published his first book on software testing in 2010 (E-mail: aysh@zuj.edu.jo).