electronics

MDPI

Article

Harnessing FPGA Technology for Energy-Efficient Wearable Medical Devices

Muhammad Iqbal Khan * and Bruno da Silva *

Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB), 1050 Brussels, Belgium
* Correspondence: muhammad.iqbal.khan@vub.be (M.I.K.); bruno.da.silva@vub.be (B.d.S.)

Abstract: Over the past decade, wearable medical devices (WMDs) have become the norm for continuous health monitoring, enabling real-time vital sign analysis and preventive healthcare. These battery-powered devices face computational power, size, and energy resource constraints. Traditionally, low-power microcontrollers (MCUs) and application-specific integrated circuits (ASICs) have been used for their energy efficiency. However, the increasing demand for multi-modal sensors and artificial intelligence (AI) requires more computational power than MCUs, and rapidly evolving AI asks for more flexibility, which ASICs lack. Field-programmable gate arrays (FPGAs), which are more efficient than MCUs and more flexible than ASICs, offer a potential solution when optimized for energy consumption. By combining real-time reconfigurability with intelligent energy optimization strategies, FPGAs can provide energy-efficient solutions for handling multimodal sensors and evolving AI requirements. This paper reviews low-power strategies toward FPGA-based WMD for physiological monitoring. It examines low-power FPGA families, highlighting their potential in power-sensitive applications. Future research directions are suggested, including exploring underutilized optimizations like sleep mode, voltage scaling, partial reconfiguration, and compressed learning and investigating underexplored flash and hybrid-based FPGAs. Overall, it provides guidelines for designing energy-efficient FPGA-based WMDs.

Keywords: power optimization; energy efficiency; wearable medical devices; continuous physiological monitoring; healthcare

Citation: Khan, M.I.; da Silva, B. Harnessing FPGA Technology for Energy-Efficient Wearable Medical Devices. *Electronics* **2024**, *13*, 4094. https://doi.org/10.3390/electronics13204094

Academic Editor: Djuradj Budimir

Received: 9 September 2024
Revised: 4 October 2024
Accepted: 15 October 2024
Published: 17 October 2024

1. Introduction

The significance of continuous health monitoring (CHM) for timely detection and prevention of diseases is on the rise due to growing health issues resulting from modern lifestyles and an aging population [1]. Monitoring vital signs helps identify symptoms of diseases, but hospitalization for such monitoring is time-consuming, burdens healthcare systems, and provides an incomplete picture. To avoid such challenges, remote health monitoring is continuously rising with the use of wearable medical devices (WMDs) [1,2].

Rapid progress in biomedical sensors, low-power electronics, signal processing, and wireless transmission technology has increased WMD adoption. However, there are some challenges to overcome to enable sustained, continuous health monitoring [3]. WMDs have to be kept small in size to provide user mobility, which not only limits resources but also the battery size. While the size constraint of WMDs necessitates smaller battery sizes, batteries have not improved as rapidly as other electronic components and remain limited in their energy storage capacity [4]. Additionally, energy harvesters have yet to reach the level of efficiency required to overtake batteries [5]. Consequently, users often face the inconvenience of frequent battery charging or replacement, which is not only laborious and costly but also poses hazardous threats to the environment [6]. Furthermore, replacing batteries in some applications, such as implanted sensors, is unfavorable, and excessive heat generated due to energy consumption may also affect body tissues [3]. On the other hand, the use of multimodal sensors and artificial intelligence (AI) integration in WMDs

for analyzing multi-physiological data is also on the rise. Traditionally, sensors' data are transferred to the cloud for AI-based analysis. However, due to power consumption and privacy-related issues, edge intelligence has been introduced [7–9], forcing all computations to be executed on a WMD. These growing requirements and associated challenges in the context of computational power and flexibility are summarized below:

- Integrating multiple or multimodal health sensors for comprehensive physiological analysis while performing edge computing over cloud computing to address privacy and security concerns regarding personal medical data necessitates increased parallelism and computational power for wearable devices.
- Incorporation of AI techniques for real-time decision and feedback demands more computational power for accuracy of results, while the continuous evolution of AI techniques and frequent customization of WMD features for individual patients demands flexibility.

Additionally, competition among WMD manufacturers to reach users faster necessitates a shorter time to market. Such increasing and rapidly changing requirements demand not only more computational power and parallelism but also system flexibility for reconfigurability, adaptability, and scalability, along with a shorter time to market. In power-sensitive designs, designers face the challenge of meeting these requirements without incurring additional power consumption costs. Therefore, a need arises to increase the computational power and energy efficiency of WMDs by using suitable computing platforms. Computations can be executed using various hardware platforms, including general purpose processors (GPPs), graphic processing units (GPUs), digital signal processors (DSPs), microcontrollers (MCUs), field programmable gate arrays (FPGAs), and application-specific integrated circuits (ASICs) [10,11].

ASICs have traditionally been used by designers to meet the energy constraints of power-sensitive devices. They are specifically designed and optimized for particular applications, outperforming all other options in terms of both performance and power efficiency. Additionally, they have lower costs for mass production. However, despite their numerous advantages, ASICs also have some limitations:

- Lack flexibility or reconfigurability;
- Higher development cost and longer time to market.

The need to accommodate rapidly changing medical requirements and procedures, evolving AI techniques, adaptability for individual patient needs, and competition to reach the market earlier make the choice of hard-wired ASICs for WMDs riskier [12].

Similarly, low-power MCUs, while offering lower energy consumption and the flexibility of reprogrammability to adapt to evolving changes, lack the ability to provide a high level of parallelism and computational power. Typically, MCUs have fixed architectures optimized for general-purpose tasks, with predefined peripherals and processing capabilities. This can limit their performance in WMD applications, which require increased parallelism and computational power due to the incorporation of AI and multimodal sensors and the need to perform all computations at the edge for privacy and security reasons. On the other hand, GPPs and GPUs provide extremely good computational power, but their energy consumption is not well suited for small battery-powered devices.

While both MCUs and ASICs offer advantages in power efficiency and cost-effectiveness, making them suitable for most power-sensitive applications, there exist some applications which require higher computational power, parallelism, flexibility in reconfigurability, scalability, adaptability, and a faster time to market all at once. Therefore, researchers are also exploring other energy-efficient, reconfigurable solutions to adapt to the rapid evolution of requirements and speed up the time to market.

FPGAs, despite being a costly choice for mass production, offer a feasible solution by providing the necessary computational power and parallelism, along with the flexibility of reconfigurability to adapt to changing requirements. They are more flexible than ASICs, more powerful than DSPs and MCUs, and more energy-efficient than GPUs and GPPs [13].

Some FPGAs even allow real-time reconfiguration to avoid downtime. However, FPGAs are generally not as power-efficient as low-power MCUs and ASICs. Therefore, power optimization strategies must be employed to make FPGA power consumption acceptable for those applications where MCUs and ASICs are not feasible due to the required computational power or reconfigurability and time to market. By optimizing power usage, FPGAs can become a viable option for these demanding applications, balancing the need for high performance and flexibility with acceptable power consumption.

Numerous recent and ongoing research efforts have focused on addressing the power consumption of WMDs using FPGAs. This paper reviews the power optimization strategies employed by different researchers in FPGA-based WMDs. The outcomes of this work are listed below:

- Identifying power optimization strategies used in FPGA-based WMDs for physiological monitoring;
- Categorizing these strategies to provide a holistic view of system design for energy efficiency, along with their benefits and constraints;
- Highlighting the shortcomings while proposing future research directions and guidelines.

Section 2 introduces continuous health monitoring using WMDs along with the exploration of low-power FPGA families from different manufacturers. Section 3 outlines the motivation behind this review by exploring existing review papers and highlighting their strengths and shortcomings in comparison to our work. Section 4 lists the objectives of this review, along with the database search methodologies and the screening of articles against the inclusion and exclusion criteria. In Section 5, the power optimization strategies found in the selected literature are analyzed and categorized along with their benefits and limitations. Section 6 presents critical analyses of the practical implications of these strategies, shortcomings, future research, and guidelines for WMDs developers. Finally, we conclude our work by summarizing the key findings and suggestions for future research in Section 7.

2. Background

Chronic diseases are the most widespread non-communicable health issues, leading to the highest mortality rates worldwide [14,15]. Many people live with complex health conditions which significantly impact their quality of life. Additionally, the occurrence of these chronic diseases increases with age. Delays in diagnosing and treating these illnesses further deteriorate patients' overall well-being, which results in frequent hospitalization and burdens healthcare systems [7]. This highlights the critical role of continuous health monitoring in the early detection of chronic disease symptoms.

2.1. Continuous Health Monitoring

Continuous monitoring of a patient's health-related parameters over time helps physicians to identify trends in health deterioration and make informed decisions. The most commonly monitored vital signs include pulse rate or heart rate, respiration rate, blood pressure, blood oxygen level (SpO2), blood glucose level, and body temperature. These vital parameters are tracked by capturing their associated physiological signals and processing them through signal processing algorithms. The relevant physiological signals are electrocardiogram (ECG), photoplethysmogram (PPG), electroencephalogram (EEG), electromyogram (EMG), electrooculogram (EOG), galvanic skin response (GSR), ballistocardiograph (BCG), and electrogastrogram (EGG). These are time series signals which are generally sparse in nature or can be transformed into a sparse domain [16–18]. The sampling rates associated with these signals are also usually relaxed [19]. Regarding usability, continuously acquiring these signals in hospitals using fixed devices is costly. Therefore, it is necessary to use wearable medical devices which patients can comfortably wear outside of hospitals.

2.2. Wearable Medical Devices

The key features of WMDs for physiological monitoring include portability, which requires battery power. Additionally, the smaller form factor of WMDs limits the battery size, necessitating lower power consumption from various processes. The most common processes performed by WMDs are shown in Figure 1. Vital signs are captured using sensors which transduce physiological activity into corresponding electrical signals. These analog signals undergo signal conditioning and filtering before being converted into digital signals through the analog-to-digital conversion process. Afterward, the data are segmented, reshaped, and formatted via preprocessing for further analytical processing. Various algorithms, including AI models, are used for analysis. Based on the results, feedback is provided. Additionally, the raw data or final results are stored locally in the WMD or transmitted to another device for further storage or detailed analysis.

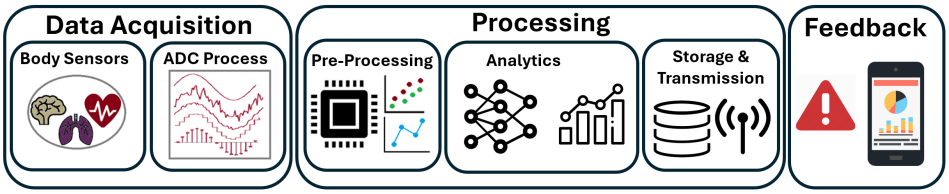

Figure 1. Typical processes in wearable medical devices.

All of the aforementioned processes consume power, which can be categorized mainly into *data acquisition*, *data processing and analysis*, and *data transmission* [3,20]. Data acquisition involves collecting data from various sensors, where the continuous operation of these sensors consumes significant power. The collected data are then processed and analyzed to extract meaningful information, which involves complex computations and can be highly power-intensive, especially if the device needs to provide real-time feedback. The data are often transmitted to other devices, such as a smartphone or a cloud, for further analysis or storage. This transmission consumes a considerable amount of power. Each of these processes is crucial for the functionality of wearable medical devices, but they also contribute to the overall power consumption, necessitating efficient power optimization strategies in WMDs.

On the other hand, the increasing demand for secure, real-time, comprehensive physiological monitoring has led to the incorporation of multimodal sensors and AI techniques, along with a shift toward edge computing. These factors are collectively increasing the demand for computational resources, making FPGAs, known for their high computational power and energy efficiency, an interesting choice to address this challenge.

Notice, however, that the terms energy and power are often used interchangeably in the literature to describe power consumption. Additionally, various phrases are found in the literature describing the reduction in power consumption, such as *power optimization*, *energy optimization*, *low-power techniques (LPTs)*, and *energy optimization strategies*. In this work, we have most frequently used the phrases *energy optimization strategies* and *LPTs*, describing the same concept.

2.3. FPGAs in Wearable Medical Devices

Wearable medical devices require efficient power management to ensure long battery life and reliable performance. FPGAs offer unique opportunities for power optimization due to their reconfigurability, customizability, and granular control. However, selecting the right FPGA family and implementation technology is crucial. This section explores various FPGA families and implementation technologies optimized for low-power applications, particularly in wearable medical devices.

2.3.1. FPGA Architecture

Although the architecture varies across vendors, the basic logic block generally includes lookup tables (LUTs) and flip-flops (FFs). These components form logic elements

(LEs) or logic cells (LCs), which in turn form configurable logic blocks (CLBs) or adaptive logic modules (ALMs), depending on the vendor's terminology. Additionally, the architecture includes configurable I/O blocks and programmable interconnects [21]. LUTs are the fundamental units within an FPGA, performing combinational logic by storing precomputed results for various input combinations, while FFs are used for sequential architectures. These logic elements are housed within LEs or LCs, while multiple LEs or LCs are organized into slices, which serve as subdivisions within a larger block. Finally, these slices are grouped together within a CLB or ALM, the primary building block of an FPGA [22]. CLBs or ALMs contain the necessary components to implement the logic functions specified by the user, forming a hierarchical structure which allows for highly flexible and configurable digital circuit implementations. These architectures also feature clock circuitry to distribute clock signals to each logic block.

2.3.2. FPGA Technologies

FPGAs are generally categorized into anti-fuse, SRAM (Static random-access memory), flash, or hybrid (SRAM + Flash) based on their implementation technology.

Antifuse FPGA: Anti-fuse FPGAs are non-volatile, meaning they retain their configuration without needing power. These FPGAs typically consume less power, making them energy-efficient. However, anti-fuse FPGAs cannot be reconfigured once programmed, limiting their flexibility. They are often used in applications requiring security against the theft of configuration data.

SRAM FPGA: SRAM-based FPGAs are highly flexible and reprogrammable, allowing numerous reconfigurations even during operation. This adaptability makes them suitable for a wide range of applications. These FPGAs store their configuration data in static RAM cells and require external memory to hold the configuration bits, which are transferred upon powering up. Their volatile nature necessitates reprogramming every time they are powered on, leading to higher power consumption. They also have higher static consumption due to leakage currents.

Flash FPGA: Flash-based FPGAs, on the other hand, use non-volatile memory cells, retaining their configuration data even when powered off. They combine the advantages of both SRAM and anti-fuse technologies. They are non-volatile like anti-fuse FPGAs and reprogrammable like SRAM FPGAs, offering flexibility, instant-on capability, and lower power consumption, with reduced static power due to minimal leakage.

Hybrid FPGA: Hybrid (SRAM + Flash) FPGAs combine the advantages of both SRAM and flash technologies by providing on-chip non-volatile configuration memory. This way, they offer the flexibility of SRAM with the non-volatility of flash.

2.3.3. Low-Power FPGA Families

The power consumption of FPGAs is primarily due to static and dynamic power consumption. Static power consumption occurs when the FPGA is powered on but not performing any active tasks, mainly due to leakage currents and depending on the underlying technology. Dynamic power consumption is associated with the switching activity within the FPGA and varies significantly based on clock frequencies and application complexity. To reduce power consumption, some manufacturers focus on technology shrinkage, while others have utilized flash-based technology to avoid the power consumption associated with SRAM technologies. Additionally, some have adopted a hybrid approach by incorporating on-chip flash configuration memory in SRAM-based FPGAs.

AMD (Xilinx) Spartan-7 and Artix-7 Families: These families are known for their optimal balance between performance and power efficiency. These FPGAs utilize the 28 nm high performance-low power process, which incorporates high-K metal gate technology to significantly reduce transistor leakage by 50% compared with previous generations. This process supports voltage scaling with two operating modes—high-performance mode at a core voltage of 1.0 V and low-power mode at 0.95 V—resulting in 10% dynamic power savings and up to 40% static power savings. This technology shrinkage allows

for reductions in both the core voltage and capacitance, achieving 40% dynamic power saving when the core voltage is scaled to 0.95 V. Additionally, power gating extends to block RAMs, reducing the overall static power, as block RAM leakage accounts for nearly 30% of total device leakage. Designers can also gate clocks at global, regional, and local resources using a clock enable, which helps reduce switching power loss in the logic core. Furthermore, the series offers partial reconfiguration capabilities, allowing users to time-slice FPGA portions to fit more IPs into smaller devices, thereby saving both power and cost. A detailed comparison is given in Table 1.

Intel (Altera) Cyclone-V and Max-10 Families: Cyclone-V and Max-10 FPGAs are designed for low-power and cost-sensitive applications, making them suitable for wearable devices. The Max-10 series utilizes 55 nm technology with an embedded SRAM + Flash process, resulting in a maximum power consumption of approximately 1.362 W for the Max10M08 device, which provides ample logic resources. It also includes a sleep mode to significantly reduce standby power and is suitable for edge computing. In contrast, the Cyclone-VE series is based on 28 nm low-power process technology, which reduces power consumption by 40% compared with the previous Cyclone-IV generation. This series also supports partial reconfiguration and includes numerous hard intellectual property (IP) blocks. Further details are provided in Table 1.

Lattice iCE40 and MachXO2 Families: These FPGAs are tailored for low-power applications. The iCE40 series, known for its use in portable and battery-powered devices, is constructed using 40 nm SRAM technology and includes on-chip non-volatile memory for configuration storage. The iCE40 UltraPlus model is capable of implementing neural networks for pattern recognition, enabling continuous intelligence at the edge. It typically consumes 1–10 mA of active current. It offers three power-saving modes—standby, sleep, and power off—which help extend battery life by allowing the application processor to enter sleep mode. The iCE40-LP family also includes two low-power modes (iCEGate and PLL) to address both static and dynamic power needs, making it suitable for ultra low-power mobile applications like handheld devices. On the other hand, the MachXO2 family is built on 65 nm technology and supports an ultra low-power standby mode. In this mode, various subsystems can be configured to shut down automatically or enter a low-power state to conserve energy. Further comparisons are provided in Table 1.

Microchip IGLOO2 and IGLOO Families: Microchip's IGLOO2 and IGLOO Nano FPGAs utilize flash-based technology, resulting in extremely low static power consumption. This flash-based architecture eliminates the need for external configuration memory, allowing the device to be instantly operational and retain its configuration even when powered off. Both devices are equipped with an ultra low-power static mode known as the flash-freeze power mode, which allows the device to enter and exit this state in under 1 µs, consuming nano power. This technology streamlines power management by avoiding the need to turn off voltages, I/Os, or clocks at the system level, while I/Os can maintain their states during flash-freeze mode. The IGLOO2 device, built on a 65 nm process, features a flash-freeze mode, while the IGLOO Nano device, built on a 130 nm process, also offers an additional sleep mode. Further details are listed in Table 1.

GOWIN Semiconductor and QuickLogic Low-Power Families: Apart from these renowned manufacturers, other vendors like GOWIN Semiconductor and QuickLogic also offer low-power FPGAs. The GOWIN GW1NZ series FPGAs are non-volatile flash-based devices known for their ultra-low power consumption, instant-on capability, and suitability for mobile and wearable applications. They also feature a low-power mode which operates at a core voltage of 0.9V. In contrast, QuickLogic's PolarPro-3 FPGAs utilize SRAM technology, offering high flexibility and reprogrammability, making them ideal for battery-powered applications. These FPGAs achieve ultra-low static power consumption of about 55 µA, enhancing the battery life of devices. Both series are optimized for low power, but they leverage different technologies to meet their power efficiency goals.

Table 1. Comparison of low-power FPGA families in terms of technology, static power consumption, and logic resources.

Manufacturer	FPGA Family	Device	Process Node	Technology	Logic Resources	RAM (kbits)	Math Blocks	Core Voltage	Static Current	Features
AMD (Xilinx)	Spartan-7 [23,24]	XC7S6	28 nm	SRAM	6000× LCs	180	10× DSP Slices	0.95 V/1.0 V	32 mA [1]	Lower power consumption than previous generation.
	Artix-7 [23,25]	XC7A12T	28 nm	SRAM	12,800× LCs	720	40× DSP48E1	0.95 V/ 1.0 V	43 mA [1]	Partial reconfiguration, voltage scaling, power gating, and clock gating.
QuickLogic	PolarPro3 [26,27]	–	–	SRAM	1019× LCs	73	-	1.2 V	55 µA [2]	Targets low power and low latency. Suitable as small CPU cores in IoT.
Intel (Altera)	CycloneV [28]	5CEBA2	28 nm	SRAM	25,000× LEs	1760	25× DSP Blocks	1.1 V	32 mA [4,**]	Partial reconfiguration; 40% lower power consumption compared with Cyclone-IV.
	MAX10 [29,30]	10M08DC	55 nm	Hybrid (SRAM + Flash)	8000× LEs	378	24× Mul (18 × 18)	1.2 V & 2.5 V	3.9 mA [1,**]	Sleep mode for standby power reduction. Suitable for edge computing.
Lattice Semiconductor	iCE40 UltraPlus [31]	UP5K	40 nm	Hybrid (SRAM + Flash)	5280× LUTs	120	8× DSP Blocks	1.2 V	75 µA [2]	Have on-chip non-volatile configuration Memory. Active current <10 mA for most apps. Three low power modes: standby, sleep, and power off.
	iCE40 LP [32]	LP8K	40 nm	Hybrid (SRAM + Flash)	7680× LUTs	128	-	1.2 V	250 µA [2]	Features 2 low-power modes (iCEGate and PLL).
	MachXO2 [33]	LCMXO2-7000ZE	65 nm	Hybrid (SRAM + Flash)	6864× LUTs	240	-	1.2 V	189 µA [2]	Has internal flash memory mimicking a non-volatile device. Standby mode for low power consumption.
Microchip Technology	IGLOO2 [34–36]	M2GL005	65 nm	Flash	6060× LEs	703	11× Mul (18 × 18)	1.2 V	1.4 mA [2,*] 6.2 mA [2,$]	Flash-dreeze mode: consumption < 1 mW.
	IGLOO nano [37,38]	AGLN250	130 nm	Flash	3000× LEs	36	-	1.2 V	20 µA [2,*]	Three low-power modes: flash, freeze, and sleep. Flash-freeze power mode has power consumption of 24 µW.
GOWIN Semiconductor	LittleBee [39]	GW1NZ-ZV2	55 nm	Flash	2304× LUTs	70	-	0.9 V/1.0 V	120 µA [3]	Offers low-power mode and normal mode. Low cost and low power; suitable for mobile and wearable devices.

[1] At 0.95 V; T_j = 85 °C. [2] At 1.2 V; T_j = 25 °C. [3] At 0.9 V, 25 °C; [4] At 1.1 V; T_j = 25 °C; * Flash-freeze mode. ** From Intel EPE tool [40]. [$] Normal mode.

Table 1 provides a detailed comparison of various low-end, low-power FPGA families, focusing on their logic resources and static power consumption. It has been observed that SRAM-based FPGAs focus on shrinking process technology to reduce dynamic power consumption. However, their static power consumption is increasing due to sub-threshold leakage and quantum tunneling. Despite this, the reprogrammability of SRAM technology allows for partial reconfiguration, which can be beneficial in delaying deployments until necessary, keeping unused logic powered off to lower power consumption. Additionally, they offer voltage scaling opportunities. For example, in the Artix-7 series, the voltage can be scaled down from 1.0 V to 0.95 V, resulting in 10% dynamic power savings and up to 40% static power savings. On the other hand, it has been observed that flash-based FPGAs have extremely low static current consumption and offer flash-freeze modes for further standby current savings. Some FPGA families combine the features of both SRAM and flash technologies, providing on-chip non-volatile memory along with SRAM fabric in hybrid mode. Devices from Lattice Semiconductor utilize this technology to offer the non-volatility of flash-based FPGAs along with the reconfigurability of SRAM-based FPGAs. It is noted that hybrid and flash-based FPGAs exhibit lower static consumption compared with SRAM-based FPGAs, along with a reasonable amount of logic resources.

The observation that various manufacturers offer different low-power families of FPGAs, each with a distinct range of power consumption characteristics, provides motivation for this literature review. This review aims to investigate the application of FPGAs in wearable medical devices, explore additional strategies for optimizing FPGA-based designs to enhance the energy efficiency, and examine existing review works on this subject.

3. Related Work and Motivation

The use of FPGAs by researchers for physiological monitoring in wearable devices is increasing to meet the demand for higher computational capabilities, as shown in Figure 2. We evaluated various survey and review studies related to the objective of this work.

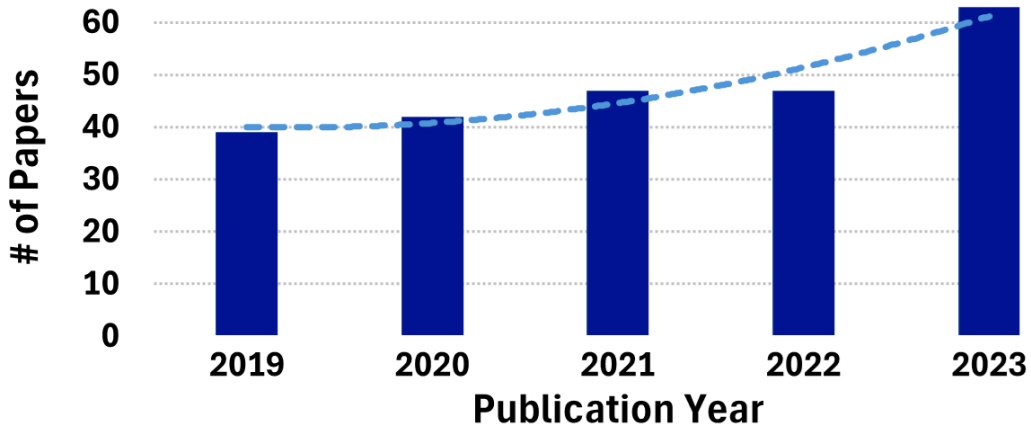

Figure 2. Number of papers published on FPGA-based wearable medical devices in last 5 years.

It has been observed that several notable studies have reviewed energy optimization strategies, with a particular emphasis on micro-controllers and CPU-based systems, while some other research has investigated these optimizations more broadly without centering on a specific platform. For instance, the authors of [41] comprehensively reviewed low-power techniques (LPTs) through a literature review for WMDs and proposed a taxonomy. However, their work does not address FPGA-based wearable devices and corresponding LPTs. Similarly, in [42], the authors reviewed the various energy-efficient strategies to minimize power consumption in the Internet of Wearable Things (IoWT) without focusing on a particular platform. The authors did not mention any FPGA-specific optimization technique at all. In a similar fashion, the authors of [43] comprehensively reviewed LPTs through a literature review for WMDs and proposed a taxonomy. However, their work does not address FPGA-based WMDs or their corresponding LPTs.

Furthermore, in [44], the authors conducted a review of various edge computing architectures in the healthcare domain. These architectures encompass operations like authentication, encryption, classification, and prediction. While the paper briefly mentions the energy optimization topic, it does not delve into a detailed discussion of LPTs and their underlying principles. Additionally, FPGA-specific LPTs were beyond the scope of this survey. In another approach, various energy-efficient solutions for IoWT applications were examined in [45]. The authors discussed common power optimization strategies. However, the paper did not review the energy-efficient utilization of FPGAs in such applications.

Likewise, the authors of [46] conducted a review on energy optimization for edge computing in mobile devices. While they touched upon several aspects of energy optimization at the device, application, and transmission levels, their primary focus remained on energy-efficient computation offloading and the corresponding execution mechanisms across various cloud environments. Notably, the review did not delve into application-specific optimization strategies for FPGA-based WMDs in physiological monitoring. In another work [47], the challenges faced by wearable devices, including energy efficiency, were reviewed and discussed. However, this work did not delve into energy optimization techniques, their classification, or the use of FPGAs in wearables. Furthermore, the authors of [48] classified energy optimization strategies into hardware (HW) and software (SW) levels. However, FPGA-based approaches were not explored. Last but not least, the authors of [49] comprehensively documented and classified energy optimization strategies, along with different architectural approaches. While their work is commendable, it primarily focuses on human context recognition applications and does not delve into FPGA-based optimization.

On the other hand, some reviews and surveys briefly mentioned energy optimization approaches. However, they did not conduct a thorough investigation or discussion of these strategies and their corresponding implications, as this was not the main focus of their studies. For example, in [50], the authors reviewed the implementation of real-time machine learning (ML) algorithms on FPGAs and systems-on-chips (SoCs) within the biomedical domain. They investigated how different researchers optimize their ML models to overcome resource limitations for a given accuracy. While the primary focus was on the trade-off between accuracy and resource utilization for ML algorithms, the authors also touched upon the impact of this trade-off on energy consumption in various instances. However, a detailed exploration of LPTs remained outside the scope of this paper. Similarly, the authors of [51] summarized a few power optimization strategies for FPGAs, with a primary focus on dynamic voltage and frequency scaling (DVFS), power gating, and clock gating. While the authors briefly mentioned "algorithm-level techniques", they did not delve into the details. Broad-level exploration of LPTs was out of their scope. Likewise, in [52], the authors reviewed various vital health parameters and their corresponding physiological signals, as well as the design considerations for smart wearable healthcare devices. It does not present LPTs.

It was also observed that some of the studies concentrated on reviewing energy optimization through improvements in FPGA architecture, technology, and associated CAD tools. While some studies briefly mentioned system design level optimization, they fell short of providing a comprehensive survey with optimization of processing algorithms, communication protocols, data acquisition methods, etc. For illustration, in [53], the authors broadly examined power optimization strategies in general across various hierarchical levels of design, including architecture, circuitry, devices, and systems. While the focus of their work was technological, architectural, and CAD tool improvements, they briefly listed and discussed some optimizations at a system-design level. In contrast, our research provides a comprehensive review of LPTs at the system design level, specifically within the wearable healthcare domain. Similarly, the authors of [54] mainly reviewed power optimization techniques and strategies for FPGAs at the technology, architecture, and CAD tool levels, targeting Internet of Things (IoT) devices. However, they did not review the optimization strategies at the system design level.

In contrast, some researchers explored energy and power optimization opportunities without a specific emphasis on WMDs. Therefore, such studies missed out on potential optimizations which could have taken advantage of the unique features of physiological signals for applications tailored to WMDs. For example, the authors of [55] reviewed energy optimization in edge computing from hardware architecture to the application layer in general and its broader prospects. However, application-specific optimization strategies for WMDs in physiological monitoring remained out of their main scope.

Similarly, the authors of [56] surveyed techniques related to optimizing and implementing convolutional neural networks (CNNs) on FPGAs. They categorized the surveyed work to emphasize both differences and similarities. While the paper briefly mentions the impact of some acceleration techniques on energy efficiency, it lacks a detailed discussion of LPTs and their underlying principles. Additionally, the focus on WMDs as potential applications remained out of the scope of this paper. In a similar fashion, the authors in [57] examined recent research on implementing deep learning models in FPGAs. They compared different topologies and development strategies in terms of performance and energy efficiency. However, the paper did not delve into the classification of LPTs specifically related to wearable physiological monitoring; that topic was beyond the scope of their study. In another work, FPGA-related LPTs within the domain of wireless sensor networks for IoT applications were explored in detail [13]. While these LPTs can be adapted for use in WMDs, it is important to note that this paper does not delve into the specifics of such utilization. In contrast, the review presented in [58] is limited to communication-related optimizations only.

There are several other review papers on energy-efficient health monitoring at the edge. However, their main focus is on allied technologies rather than the computing platform. For example, some papers focused on low-power wireless transmission technologies suitable for power-sensitive WMDs, such as Bluetooth Low Energy (BLE), near-field communication (NFC), and radio frequency identification (RFID) [59–61]. BLE is designed for low power consumption, making it ideal for wearable medical devices. Similarly, NFC enables data exchange between devices in close proximity, typically within a few centimeters. It is used for secure patient identification, device authentication, and data transfer, allowing healthcare providers to quickly access patient information by tapping an NFC-enabled device to a patient's wristband [61]. On the other hand, RFID uses electromagnetic fields to automatically identify and track tags attached to objects. Its applications range from low-power physiological health monitoring [59] and medication tracking to structural health monitoring [62–64] due to its energy efficiency. Additionally, some review papers focused more on energy harvesting approaches rather than exploring algorithm-based optimization strategies [65].

While the above-mentioned reviews and surveys, summarized in Table 2, provide notable overviews within their respective scopes, our review paper addresses the identified gaps. We offer an exhaustive list and in-depth analysis of cutting-edge optimization techniques and strategies, including the underlying principles, benefits, and corresponding implications for FPGA-based WMDs. To the best of our knowledge, this is a more detailed review of its kind which addresses the power optimizations specific to FPGA-based WMDs for physiological monitoring.

Table 2. Comparison of review papers on FPGA-based, energy-efficient wearable medical devices for physiological monitoring.

Ref.	Year	Focus	Wearable Devices	Healthcare Applications	Common LPTs	FPGA-Specific LPTs	Categorization of LPTs	Limitations
[41]	2024	Wearables in medical domain	●	●	●	○	●	Comprehensive review of LPTs for WMDs and taxonomy. However, FPGA-based wearables and corresponding LPTs were not focused upon.
[52]	2024	Low-power wearables	●	●	○	○	○	This paper is focused on general design consideration for smart wearable healthcare devices; it does not present LPTs for WMDs.
[50]	2024	Machine learning (ML) in biomedical field	◐	●	◐	◐	○	This work is focused on optimization of ML models to overcome resource limitations for a given accuracy. However, a detailed exploration of LPTs remained out of scope.
[51]	2024	Wearable and Implantable Devices	●	●	◐	◐	◐	The main focus is on dynamic voltage frequency scaling, power, and clock gating along with brief mention of "algorithm-level techniques". Broad-level exploration of LPTs out of scope.
[53]	2023	General use of FPGA	○	○	◐	●	○	While the focus of their work lies on technological, architectural, and CAD tool improvements, they briefly listed and discussed some optimizations for system-level design. Comprehensive review of LPTs at system-level design, specifically within WMDs, is missing.
[42]	2023	IoWT in healthcare	●	●	●	○	○	Reviewed various energy efficient strategies in IoWT without focusing on a particular platform. FPGA-specific optimization techniques were not considered.
[43]	2022	Wearables in medical domain	●	●	●	○	●	Comprehensive review of LPTs for WMDs and taxonomy. However, FPGA-based wearables and corresponding LPTs were not under focus.
[44]	2022	Edge computing in healthcare	●	●	◐	○	○	The authors reviewed various edge computing architectures in the healthcare domain. However, a detailed discussion of LPTs and the underlying principles for FPGA-based WMDs was out of their scope.
[54]	2021	IoT	○	○	●	○	○	Reviewed LPTs at technology, architecture, and CAD tool levels, targeting IoT devices. A review of LPTs for the system design level is missing.
[45]	2020	IoWT	●	◐	●	○	●	Although various energy-efficient solutions for IoWT applications were examined, energy-efficient utilization of FPGAs in such applications was missing.
[55]	2020	Edge computing	◐	◐	●	◐	●	Comprehensively reviewed energy optimization in edge computing in broader prospects. However, application-specific optimizations for WMDs in physiological monitoring remained out of the main scope.
[46]	2020	Mobile devices	○	○	●	○	●	A comprehensive review with a primary focus on computation offloading to cloud in mobile devices. However, application-specific optimizations for FPGA-based WMDs in physiological monitoring were not the main focus.
[56]	2020	FPGA-based convolutional neural networks	○	○	◐	◐	○	A survey of techniques for optimization of CNNs on FPGAs. It lacks a detailed discussion of LPTs for WMDs as potential applications.
[57]	2019	FPGA-based deep learning (DL)	○	◐	◐	◐	○	This work examined recent research on implementing DL models in FPGAs in terms of performance and energy efficiency. However, classification of LPTs for WMDs was beyond the scope of their study.
[13]	2019	Wireless sensor networks (WSNs) in IoT	○	○	◐	●	●	Explores FPGA-related LPTs within the domain of WSN for IoT applications. However, the use of such LPTs in WMDs was not covered.
[47]	2017	Wearable devices	●	●	◐	○	○	Does not delve into the energy optimization techniques, classification, or the use of FPGAs in wearables.

Table 2. Cont.

Ref.	Year	Focus	Wearable Devices	Healthcare Applications	Common LPTs	FPGA-Specific LPTs	Categorization of LPTs	Limitations
[49]	2017	Human context recognition (HCR) via WSNs	●	●	●	○	●	A comprehensive review with a primary focus on HCR applications. However, it does not delve into FPGA-based optimization.
[58]	2016	Remote healthcare monitoring using IoT	●	●	◐	○	○	Limited to communication-related optimizations only.
[48]	2015	Wearable devices	●	●	●	○	●	Classifies energy optimization strategies into HW and firmware levels. However, FPGA-based approaches were not explored.
This Work		Wearable medical devices	●	●	●	●	●	–

● indicates that the subject review work covered this topic; ◐ indicates that the subject review work partially addressed this topic as compared to our work. ○ indicates that the subject review work did not cover this topic. The term LPTs indicates 'Low-power techniques' or 'energy optimization strategies'.

4. Review Methodology

This paper aims to review the current state-of-the-art application of FPGAs in WMDs in the context of power optimization and energy efficiency while highlighting the challenges for further research opportunities. A review of the trends in this research field in the last five years was performed to identify suitable research papers for the objective.

A comprehensive search process was run on different databases using a well-thought out set of search keywords. Web of Science, ACM-DL, IEEE-Xplore, Scopus, Science Direct, PubMed, and Google Scholar were explored using the following search keywords:

FPGA AND wearable AND (health OR healthcare OR medical OR Biomedical OR monitoring OR management OR diagnosis OR treatment OR rehabilitation)

The initial search keywords were further enhanced by including synonyms such as "*field programmable gate array*" along with terms like *portable*, *edge*, *IoT*, or *remote* and *power* or *energy*. Moreover, the health-related terms were optimized to *health** or *medical* or *disease** or *biomedical*.

We included works written in English with full text available which implemented their research on FPGAs and evaluated power optimization approaches in their systems. Works focusing on other parameters such as performance, accuracy, or resource consumption which did not directly target power optimization were excluded, even if they had some power improvement.

After identifying the relevant articles, we removed duplicates, considering the possibility of overlapping results across multiple databases. Next, we assessed the eligibility of the shortlisted papers against the above-mentioned inclusion and exclusion criteria. Finally, 52 studies were selected for the final review.

5. Power Optimization Strategies

Designing any system involves various aspects, from the preliminary conceptual design to the final hardware implementation [51]. Energy-efficient design requires a holistic approach which considers all aspects of the system. This includes selecting energy-efficient hardware, designing energy-conscious algorithms which minimize computational overhead, minimizing transmission energy cost, scheduling tasks to optimize resource usage, and leveraging the energy-saving capabilities of the hardware platform. By integrating these elements, one can ensure that the system operates efficiently, reducing overall energy consumption. Neglecting any of these aspects can compromise the system's energy efficiency, making a holistic approach essential.

In this work, we systematically present the power optimization strategies explored during our review of various FPGA-based WMDs in the related research papers. These strategies were categorized into "*Designing Energy-Conscious Algorithms*", "*Minimizing Transmission Energy Costs*" "*Energy-Efficient Task Scheduling*" and "*Exploiting FPGA Features*", as summarized in Figure 3.

5.1. Designing an Energy-Conscious Algorithm

At the heart of WMDs lies an algorithm which processes and analyzes data collected from health sensors, identifies patterns, provides predictive analysis, and offers personalized health insights. It can range from a simple set of instructions to complex AI algorithms commonly found in the reviewed literature. Examples include the infinite impulse response (IIR) filter [66], CNN [67], support vector machines (SVMs), multilayer perceptron (MLP) [67], random forest (RF), and artificial neural networks (ANNs). These algorithms consume energy primarily due to their computational tasks, such as data processing, real-time analysis, and communication. The complexity of computations increases with the algorithm's complexity, leading to higher energy consumption. Simplifying these computations and optimizing the algorithms can help reduce their energy usage.

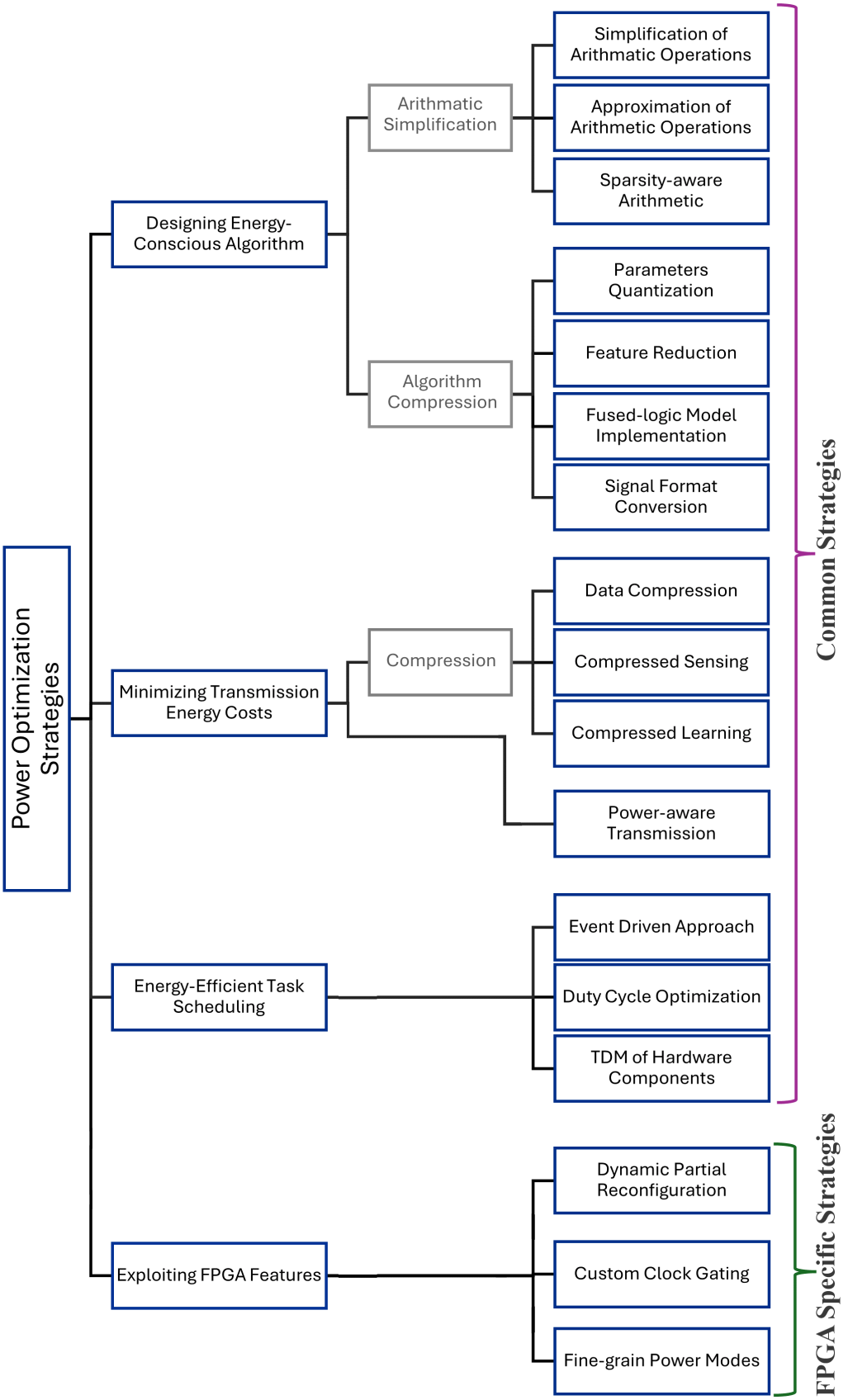

Figure 3. Categorization of identified power optimization strategies.

Wearable devices are often battery-powered and limited in computational resources, such as memory and processing power. As the complexity of computing algorithms in-

creases, so do the demands on power consumption and computational resources. However, high-performance algorithms are essential for real-time monitoring and timely interventions in medical scenarios. The incorporation of AI in WMDs has further increased this complexity [68]. Therefore, it is necessary to optimize these complex algorithms for resource efficiency, power consumption, and performance before implementing them in WMDs. Optimized algorithms ensure efficient use of resources, allowing the device to perform its tasks without overloading the system. This optimization can significantly improve performance, resulting in faster data processing and quicker response times with an acceptable level of accuracy. Most importantly, optimized algorithms can reduce the number of computations required, thereby conserving energy and extending battery life. These benefits are particularly important for continuous health monitoring tasks which require real-time data processing over long periods. To address issues of resource consumption, memory, and power usage, various optimization techniques, such as approximate computing, parameter quantization, feature reduction, and signal format conversion, are employed to compress these models [69].

5.1.1. Simplification of Arithmetic Operations

Arithmetic operations, such as multiplication and division, are resource-intensive and consume significant FPGA logic and power. These operations can often be replaced by simpler shift and add operations. Similarly, logic-wise multiplication can be substituted with XNOR operations [70,71]. These simplifications reduce computational complexity and resource consumption, leading to lower power usage.

For example, the authors of [72] binarized the weights, allowing logic-wise multiplication operations using XNOR instead of the power-consuming MAC operation. This resulted in simplified computations, thereby reducing resources and dynamic power consumption, but at the cost of a slight reduction in accuracy due to the simplification of arithmetic operations. Similarly, in [67], the authors used a binary hyperbolic tangent as an activation function to allow a logic-wise multiplication operation (XNOR) in their bCNN-based ECG classifier, resulting in significant resource and power consumption at the cost of a small degradation in accuracy. Likewise, in [69], the authors replaced the multiply–accumulate operation with bitwise XNOR operation, exploiting the fact that weights and feature maps are constrained to negative or positive one and thereby enabling one-bit operations. This significantly reduces memory access and computational resources, leading to a reduction in power consumption with acceptable accuracy.

Another approach is to replace multiply–accumulate (MAC) operations with shift–accumulate (SAC) operations [73–75]. Using shift–add operations instead of multipliers can reduce hardware complexity and power consumption because shift-and-add operations are generally simpler and require fewer resources than multiplication, though this may affect the overall speed. These simplifications significantly reduce the complexity of computations and the corresponding hardware, thereby lowering power consumption. For instance, the authors of [66] replaced the DSP block with shifters by choosing filter coefficients with a power of two, resulting in significant area and power reduction without compromising the functionality. Filters with coefficients of the form 2^n can be implemented using addition and shift operations instead of multipliers, resulting in reduced resource and power consumption [76].

Similarly, the authors of [77] converted the MAC operations with SAC. According to the authors, the dynamic power consumption of a 12 bit multiplier is approximately 13× times higher than that of the 15 bit shifter. Likewise, the authors of [78] replaced the multiplier with shift-and-add operations in their lifting-based discrete wavelet transform (LDWT) PVC recognition system. In another approach, an integer Haar wavelet transform (IHT) was used by the author of [79] for denoising the ECG signal, where division by a "2" operation was also implemented using a simple shift operation. This resulted in lower complexity and thereby lower power consumption.

5.1.2. Approximation of Arithmetic Operations

The complex functions can also be approximated with simpler ones, having similar curves provided that the accuracy degradation is within an acceptable range [80]. Likewise, an operation involving multiplication or division by a particular constant can be avoided in AI operation because it linearly scales the data, and the classifier can adjust the hyperplane with the corresponding scale.

The author of [80] exploited the fact that multiplication or division by a particular constant linearly scales the data. Removing such multiplication or division will not impact the classifier's output, as it will adjust the hyperplane with the corresponding scale. They also observed that the approximation of complex functions with simpler ones, having similar curves (e.g., natural log and square root), can also be used, provided that the degradation is within an acceptable range. On the other hand, the authors of [81] replaced conventional adders with approximate adders in order to reduce the power consumption by 21% at the cost of a 3% loss in accuracy.

5.1.3. Sparsity-Aware Arithmetic

Sparsity-aware arithmetic reduces the number of active computations by leveraging the sparsity (presence of many zero or near-zero values) to optimize the computations. This approach can significantly reduce the number of operations and memory usage, leading to more efficient processing and thereby reducing computational resources and power consumption. In [82], the authors used sparsity-aware arithmetic in a BioCNN-based emotion detector. They exploited the sparsity generated by weight pruning to avoid unnecessary computations in zero-operand cases, thereby reducing the number of computations and the power consumption to 150 mW. Sparsity-aware arithmetic not only enables faster processing times but also allows for the efficient use of FPGA resources, facilitating the deployment of larger or more complex models on the same hardware. Additionally, it can reduce latency by decreasing the number of operations per inference. However, it is crucial to carefully manage sparsity-aware techniques to prevent significant accuracy loss.

5.1.4. Parameter Quantization

Arithmetic operations with real numbers on an FPGA consume substantial power due to floating-point (FP) computations. Moreover, these FP operations utilize significant on-chip resources, memory bandwidth, and inference times [7]. Parameter quantization is used to reduce the model's size, memory footprint, and resource consumption by compromising the precision of model parameters. It involves converting high-precision floating-point numbers (e.g., 32 bit) into lower-precision formats (e.g., 8 bit integers). To optimize the model for resource-constrained hardware, the model's parameters are quantified from float to fixed for compression and efficient inference [83]. Such optimization also results in the reduction of resource consumption and improves power consumption. For instance, the authors of [84] implemented a real-time apnea detection module, taking input from a pulse oximeter and single-channel ECG sensors and using a BiNN on an Artix-7 Nexys FPGA. The input data were quantized into eight-bit binary values. The hyperparameters of the trained model were also binarized, enabling the substitution of MAC operations with SAC operations, including the use of a SAC-based activation function. Such implementation of the SABiNN model, as opposed to a feedforward neural network (FNN), resulted in a reduction in power consumption of nine times and a reduction in resource usage of four times, with only a minimal but acceptable loss in accuracy.

The authors of [68] also exploited parameter quantization in their SNN-based ECG classifier. The parameters were mapped to the $(-1, 1)$ range during the training phase and then quantized to a six-bit fixed point for hardware implementation. This approach significantly reduced complexity and power consumption at the cost of little accuracy loss from 98.39% to 98.22%. An arrhythmia classification at the edge was designed based on the DCNN model in [83], where a large-scale DCNN was optimized to accelerate its

inference on the FPGA-based embedded platform by quantizing the parameters of the pruned model. The weights and feature maps of the pruned model were quantified to unsigned 8 bit integers, while the bias vectors were quantified to signed 32 bit integers. Such integer operation resulted in a reduction in communication, along with memory and power consumption. In another approach, the authors of [77] converted the FP parameters to corresponding integer values in their FNN-based sleep apnea detector for power consumption.

Similarly, in [66], the authors used fixed-point arithmetic in their QRS extractor based on the Pan and Tompkins algorithm. The IIR filters in the digital signal processing (DSP) section of this algorithm are power-hungry components due to the large number of arithmetic computations involved. They reduced the number of bits at input from 16 to 4, thereby reducing the bit size of all of the processing elements involved in the computation. The authors of [72] used fixed 16 bit precision and also binarized the weights, simplifying computations and thereby reducing resource usage and dynamic power consumption. However, this came at the cost of a slight reduction in accuracy due to the simplified arithmetic operations. Likewise, ECG signal classification using dynamically biased LSTM was proposed in [85], where weights were quantized to fixed points for hardware implementation. The author quantized the widths from FP32 to INT4 to improve power consumption, with an acceptable loss of accuracy of 2.4%.

Likewise, the authors of [78] used fixed points (16 bit) to represent LDWT factors, computed in a parallel fashion to reduce the clock cycles from n to 1. The authors were able to achieve a 20% reduction in power consumption operating at 35 MHz with 98.29% accuracy while using only 4.36% of the total resources. The authors of [86] also quantized the parameters of each trainable layer to 8 bit integers in their 1D CNN-based ECG classifier. In another approach, an integer Haar wavelet transform (IHT) was used by the author of [79] for denoising the ECG signal. This works by using integer coefficients, thereby eliminating FP operations. Some other authors [83,87–91] also replaced the FP operation with fixed-point or integer operations to reduce resource consumption, computational load, and power consumption.

In contrast, the authors of [92,93] kept the FP arithmetic to maintain the dynamic properties of the ECG signal in their ECG denoising system. However, to reduce resource consumption and the off-chip bandwidth, they used half-precision (16 bit) FP instead of single-precision (32 bit) FP.

Quantization significantly reduces the size of a model by using lower-precision arithmetic operations, which are faster and consume less power. This process is crucial for deploying complex models on wearable devices, which are constrained by limited resources and battery life. FPGAs can be programmed to use any bit width for their operations, allowing for more precise control over the quantization process.

5.1.5. Feature Reduction

Feature reduction in AI algorithms reduces data by eliminating less relevant features while keeping essential information. Such reduction produces smaller models, simpler computations, less memory, and less data movement within the FPGA. It can enhance the performance, resource, and power efficiency of FPGA implementations in WMDs. By decreasing the number of features, the AI model becomes less complex, leading to faster processing times, which is vital for real-time monitoring [94]. However, this may reduce accuracy, and thus careful feature selection is essential. Fewer features also mean reduced memory and computational load, allowing for smaller, less powerful FPGAs and extending battery life. Additionally, lower power consumption reduces heat generation, improving user comfort.

The authors of [95] reduced the area and power consumption in their single-lead ECG processor by using a reduced feature set of four (RFS4) features. Using only four morphological features simplifies the computations, thereby reducing the computations, area, and system power consumption. In [96], the authors also used a feature reduction approach in

their low-power stress detector to reduce the computational complexity, memory access, and resources and implement a low-power design with 82 mW of consumption. The authors of [81,97] also used a similar approach, feature selection, in epileptic seizure detection for power reduction. Employing a lower number of features for prediction with a given confidence can reduce computational complexity. Additional features can be activated only when the desired confidence level is not achieved. This strategy minimizes the overall computational complexity, thereby reducing power consumption.

Channel selection [94] and weight pruning [98] are other approaches used to reduce computational load and power.

5.1.6. Fused-Logic Model Implementation

Fusion of operations across adjacent layers or fusion of adjacent layers itself combines multiple computational steps into a single step to reduce memory access and improve computational efficiency, which can reduce resource and power consumption. Such fusion of operations can increase the overall throughput and reduce the latency of the system, which is important for real-time physiological monitoring, where timely data processing is critical. However, such operations are highly optimized for a particular scenario or application, which reduces the flexibility of the system to adapt to changing requirements. The authors of [99] integrated their feature selection process with the classifier to form a single architecture for their EEG-based seizure detection system. This strategy improved performance along with a reduction in computational complexity and power consumption, resulting in 2.018 µW of power.

In another approach, the authors fused their batch normalization with convolution to reduce unnecessary computations in their PPG-based heart rate estimator [100]. The convolution operation mainly consists of multiplication and addition operations. Usually, the next layer is a batch normalization (BN) layer, where the main operations are subtraction and division, which are resource- and power-hungry. These computations can be fused with the previous layer's computations to remove repeating computations. This results in a reduction in computational complexity and can thus also reduce resource and energy consumption. The authors of [67] fused their qMLP with the bCNN model to enable an end-to-end architecture for their binary image-based ECG classifier for resource and power reduction.

The fusion of operations and layers can significantly enhance efficiency, performance, and power optimization. However, it can limit the flexibility of the system.

5.1.7. Signal Format Conversion

Some researchers have converted signal formats to reduce complex arithmetic operations in their approaches. This conversion allows them to leverage well-established, efficient models from other domains, which can extract features more effectively. For instance, conversion of an ECG signal to a binary image enabled the authors of [101] to use simple addition and shift operations in their TNN-based cardiac arrhythmia detector. This resulted in a reduction in energy per operation of three times. Moreover, this simplification reduced the bit sizes of the weights, thereby simplifying the memory controller. This not only increased the overall system speed but also reduced latency. The authors of [67] also used the same approach to convert ECG signals into binary images in their ECG classifier on a wearable edge device to exploit the benefits of the bCNN for resource and power optimization.

By transforming complex representations into simpler ones, this method also reduces memory requirements, computational load, and power consumption, which are critical factors for WMDs. However, this approach introduces the overhead of initial data processing for conversion and post-processing, which may degrade system latency. On the contrary, the authors of [102] used a simple slope-based detection (SBD) algorithm instead of complex machine learning for seizure detection. To save power, they computed the data

at shorter intervals instead of at the sampling period and determined the running sum of the squared data values before calculating the slope.

In conclusion, researchers have employed various optimization techniques to enhance the efficiency of complex algorithms in WMDs. These optimizations address the challenges posed by the increasing complexity of AI-integrated algorithms, ensuring that AI models are compressed and optimized to operate effectively within the resource, memory, and power constraints of WMDs.

The benefits and constraints associated with these optimization techniques are summarized in Table 3, along with their descriptions.

Table 3. Optimization strategies for designing energy-conscious algorithms.

Strategy	Description	Advantages	Constraints
Simplification of arithmetic operations	Complex operations, such as multiplication, are replaced with simpler ones (e.g., MAC to SAC or MUL to XNOR) to reduce computational resources.	Requires fewer logic elements and memory resources. Simpler operations can lead to lower switching activity. Reduced resource usage allows more parallelism for performance, lower latency, and faster computation. Reduced power consumption. Less heat generation leads to more comfort and safety of WMDs.	Possible loss of precision and therefore accuracy. Accumulation of smaller errors over time can affect reliability. Can limit the flexibility of the system.
Approximation of arithmetic operations	Complex functions are approximated with simpler ones, having similar curves, to reduce computation complexity.		
Sparsity-aware arithmetic	Exploiting the sparse nature of signals to reduce the number of active computations.		
Parameter quantization	Floating-point operations are replaced with fixed-point or even lower-bit integer operations (e.g., FP-32 to FP-16, INT-32, INT16, INT8, or INT4), with binarization of parameters to 1 and −1 values.		
Feature reduction	Reduces data and computations by eliminating less relevant features.		
Fused-logic model implementation	Fusion of operations across adjacent layers or across the components to reduce memory access and improve computational efficiency.		
Signal format conversion	Conversion of signal formats to reduce complex arithmetic operations (e.g., ECG to binary image).		

5.2. Minimizing Transmission Energy Costs

Continuous health monitoring involves collecting large amounts of data, which can be stored locally or transmitted to a higher level for further storage and analysis. However, transmitting continuous streams of physiological data at high data rates and frequent intervals costs energy consumption, which needs to be minimized to extend the battery life of WMDs. The energy consumption is directly proportional to the volume of data and the frequency of transmission. Consequently, high data volumes and frequent transmissions can quickly drain the battery. To address this issue, researchers have developed several approaches focused on the idea of data compression and power-aware transmission, aiming to reduce the data volume, lower the data rates, and decrease the transmission frequency, thereby minimizing energy consumption. Different data compression approaches, their energy minimization methods, and their associated advantages are further explained in the following sections and also summarized in Table 4.

5.2.1. Data Compression

Data compression at the sensor node lowers the amount of data to be transmitted and thereby also lowers the power consumption. The authors of [103] proposed a lossless compression scheme based on run-length coding for ECG signals. The signal is compressed through duplicate data encoding, exploiting the distribution and correlation of adjacent beats of the ECG. They obtained a compression ratio of 2.11:1 for lossless compression. Fusion of data from the body sensor network (BSN) also faces the challenge of higher data volumes and power consumption.

Table 4. Optimization strategies for minimizing transmission energy costs.

Strategy	Description	Advantages	Constraints
Data compression	Compress the data in lossy or lossless fashion to reduce the transmission load	Reduction in power required for transmission. Reduction in storage requirement.	Trade-off between power saved from transmission vs. spent for compression & reconstruction. Power saving vs. accuracy trade-off
Compressed sensing	Efficient acquisition and reconstruction of signals using fewer samples than typically needed	Reduction in power required for both acquisition and transmission. Reduction in storage requirement.	
Compressed learning	Focuses on directly inferring information from compressed measurements using machine learning tools	Eliminate the resource and power consumption required for reconstruction.	Additional design effort for optimizing sensing and inference with deep learning during training Trade-off: power consumption vs. accuracy
Power-aware transmission	Adjusting the transmission load based on the current battery level	Maintain availability of critical data by prolonging the battery lifetime.	Could impact real-time monitoring

In another work [104], the authors used a data compression approach in their ballistocardiograph (BCG)-based heart health monitor. The data sampled at 600 Hz with 16 bits for each sample pose a challenge for storage and wireless transmission due to power consumption in wearable applications. Modified delta encoding was used for online compression of large data volumes, achieving a compression ratio of 5.17 with negligible loss to reduce transmission power consumption and the storage requirements. In another approach, the authors of [105] proposed transmitting only lossy compressed ECG data initially until an anomaly is detected. This method achieved a compression ratio of 7.8× at the cost of an RMS error of 4.61, requiring only 18% of the power consumption compared with uncompressed transmission using a Bluetooth transceiver. In the event of anomaly detection at the receiving end, the sensor node can be prompted to send the error residue. This allows the original signal to be accurately reconstructed without any loss, enabling further analysis of the anomaly.

Depending on the compression technique used, there might be some loss of data fidelity. Any loss of data accuracy could potentially impact clinical decisions based on the monitored data. Moreover, decompression and analysis of compressed data require additional computational resources, which might complicate the device's design and operation at the receiving end.

5.2.2. Compressed Sensing

Compressed sensing (CS) [106], also known as compressive sensing or sparse sampling, is a signal processing method which efficiently captures and reconstructs signals using fewer samples than typically required, known as sub-Nyquist sampling [107]. This approach reduces the need for continuous sampling and the associated communication load, which can frequently drain the battery [49]. This technique takes advantage of the fact that many natural signals are sparse or compressible in certain domains. By utilizing fewer measurements, compressed sensing can greatly decrease the data required for accurate signal reconstruction, which in turn can significantly lower the transmission load and reduce power consumption. By leveraging this inherent sparsity in most physiological signals, CS becomes a powerful tool compared with conventional compression techniques.

The authors of [108] introduced a low-complexity hardware implementation of CS on an FPGA for reconstructing compressively sensed signals using the matching pursuit (MP) algorithm. The proposed hardware design was implemented on the programmable logic (PL) part of a Zybo board, achieving a significant reduction in computational time. It was 75× faster than the processing system (PS) implementation on the same board at the cost of extra resources of FPGA. In [109], the authors also implemented multichannel EEG compressive sensing on an FPGA to optimize power consumption in wireless sensors and enhance real-time performance. The system can collect, compress, transmit, and reconstruct EEG signals in real time. Moreover, the authors used a binary permutation block diagonal (BPBD) sensing matrix due to its low computational complexity and resource

efficiency, which helped reduce the power overhead associated with the compressive sensing processes.

Compressive sensing reduces the power consumed for data transmission by collecting data in a compressed manner. This is because fewer samples are needed to represent the signal, which translates to less data being transmitted. However, this process requires additional computations for both compression at the front end and reconstruction at the receiving end, which in turn demand extra resources and power. Therefore, there is a trade-off between the power saved from transmitting less data and the power spent on compression and reconstruction. Moreover, if the data are not perfectly sparse, or if there is noise, then the process of reconstruction can introduce errors, leading to inaccuracies in the final analyses.

5.2.3. Compressed Learning

In contrast to the CS approach, which has reconstruction overhead at the receiving end, compressed learning (CL) focuses on directly inferring information from compressed measurements using machine learning tools [110]. This approach mitigates the reconstruction overheads and associated power consumption. It combines signal processing and machine learning to infer information from a signal using a small number of measurements. It leverages deep learning techniques to optimize both the sensing and inference processes during the training phase. Moreover, it also eliminates the need for additional computations, resources, time, and power required for the reconstruction process. The authors of [111] proposed a compressed learning approach in their abnormal ECG detection system. In the CL approach, the features are directly extracted from compressed data while ensuring acceptable levels of accuracy. CL reduces the amount of data required to be stored and processed, thereby also reducing storage, memory accesses, and computations, which is highly important in WMDs.

Similarly, the authors of [112] proposed a CL method for on-device ECG signal classification, which operates directly within the compression domain, eliminating the need for reconstruction. This approach achieved a 2.6× reduction in model size at a compression ratio (CR) of 0.2, with only a loss of 0.028 in the macro F1 score compared with the uncompressed scenario. It significantly reduced the dimensions of raw data, thereby decreasing the required processing power. In another instance [113], the authors also performed feature extraction and classification directly on the compressed data in their low-power seizure detection processor. All of the computations are performed on the sensor node, and only the 1 bit result is transmitted. This significantly reduces the transmission load and power consumption. Likewise, the authors of [20] employed a similar approach to analyze compressed data received from the sensor node without reconstruction to detect anomalies. They were able to achieve acceptable accuracy, with a compression ratio of 16:1 and a significant reduction in transmission power.

CL is a relatively new approach which can significantly reduce the transmission power at the sensor node and computation power at the receiving node. However, the accuracy of the results can be affected by the quality of the compressed measurements.

5.2.4. Power-Aware Transmission

Power-aware transmission refers to techniques which adjust a device's data transmission based on the current battery level to optimize power consumption. If the battery depletes unexpectedly, then critical physiological data might not be transmitted or recorded, potentially compromising patient monitoring and care. To mitigate this, the device can reduce the frequency or amount of data transmitted when the battery level is low, prioritizing critical data. For example, power usage can be adjusted by sending compressed data instead of raw data, ensuring that essential monitoring continues even with a low battery.

The authors of [105] proposed a similar approach for "power-aware transmission mode" implementation, where the ECG data are compressed through a lossy approach while the residual error is stored in local memory in an entropy-encoded form. The original

signal can be fully represented losslessly by combining the lossy compressed signal with the encoded residuals. The system can switch to a lossy transmit mode if the battery level is below some threshold. In the event of anomaly detection at the receiving end, the sensor node can be requested to send the error residue. This allows the original signal to be accurately reconstructed without any loss, enabling further analysis of the anomaly.

The authors of [114] exploited a similar approach in their ECG monitoring system. A dual transmission mode strategy was employed, depending on the battery level. The transmitter within the sensor node continuously transmits raw ECG samples when the battery voltage is above a predefined threshold (referred to as the high-power mode). However, when the battery level falls below the threshold, an energy-saving mode is activated, and only critical information is sent in encrypted (3 bit) form. This helped induce a 98% reduction in transmission load and therefore power consumption compared with the high-power mode.

However, such implementations require more resources in FPGAs and impact the device's real-time monitoring capabilities due to reduced data transmission rates. Nevertheless, these downsides are often offset by the benefits of preventing the loss of critical information, extending device life, and improving reliability.

The benefits and constraints of these optimization techniques are summarized in Table 4 along with their descriptions.

5.3. Energy-Efficient Task Scheduling

Strategic scheduling of tasks and components to leverage specific hardware features can significantly reduce power consumption. In physiological monitoring, some tasks rely on the successful completion of other tasks. For example, detecting an anomaly in an ECG signal can trigger a subsequent arrhythmia classification step. Conversely, some tasks require only periodic activation, such as turning on the LED of a PPG sensor. Significant efforts have been made by different authors to optimize power consumption by scheduling tasks in a way which selectively activates only the necessary components at any given time. They employed various approaches, including event-driven activation, duty cycle optimization, and time sharing of hardware components using time division multiplexing (TDM). All of these approaches rely on the idea of minimizing unnecessary active components or their active time to reduce energy consumption.

5.3.1. Event-Driven Approach

Designs based on an event-driven approach involve activating hardware components only in response to specific events rather than running them continuously. This can significantly reduce energy consumption by ensuring that components are only activated when necessary, thus avoiding wasteful energy expenditure during idle periods.

The authors of [72] proposed a two-stage ECG classifier using an event-driven approach. The idea behind this is the observation that most heartbeats are usually normal. The algorithm is implemented on a Zynq XC7Z020 FPGA, where the first stage only differentiates between normal and abnormal classes. The Softmax function normalizes the output of the first stage. If the classification score exceeds some preset value, then the classifier sends the normal class as the output; otherwise, the second stage will be triggered to further classify the abnormal class into different types of arrhythmia. Another approach implemented a three-level event-driven mechanism for arrhythmia classification on a Xilinx XC5VLX110T [115]. The ECG signal, converted through LC-ADC, is segmented into independent heartbeats by the QRS locator module. The correlation information among heartbeats is then reproduced by the beat segment refactor module. This reconstructed information is then further classified using an abnormal heartbeat detector module. Upon detection of an abnormal heartbeat, the cardiac arrhythmia classifier is awakened to classify the heartbeat further; otherwise, it remains dormant. The authors reported a 44% reduction in power consumption when operating in detection mode compared with classification mode.

A neuromorphic processor was proposed on an event-driven based architecture, where a spiking neural network (SNN) is used with a leaky integrate-and-fire (LIF) model of neurons for ECG classification [116]. At every time stamp, the potential is integrated by an input spike and is also lowered by a leaky term. When the integrated value of potential reaches a threshold, a firing event happens. In the proposed processor, an event handler is designed for each of the three phases of the LIF model. The event controller triggers an event handler based on the timestamp and ID of the neuron received in the processor via the bus. These event handlers are triggered based on the conclusion of the previous event. The processor remains in an idle state for the rest of the time. They were able to reduce power consumption by 21.69% and resources by ≈15% at the cost of reduced accuracy of 83%. The authors of [117] also used an event-driven concept in epileptic seizure detection, dividing the process into monitoring and detection stages. The monitoring stage extracts only one feature to avoid redundant computations and operates in both inter-ictal and ictal states. The detection stage remains powered off and is only activated temporarily by the controller when a detection event happens in the monitoring stage. It extracts more features to improve accuracy and thus reduce false positive alarms in the first stage. They were able to reduce dynamic power consumption to 1 μW with a trade-off between detection speed and false positive detection.

A similar approach was also used in [114], where upon detecting a low battery event, the transmission mode was switched from transmitting complete raw data to sending only critical data to conserve power. In case an anomalous event is detected in ECG data, the system can revert back to transmitting the complete raw signal for detailed analysis at the receiving end. This significantly reduces the transmission load and power consumption. In [20], the authors also implemented an event-driven signal reconstruction module. Their system is divided into three modules: compression, analysis, and reconstruction. The reconstruction module remains dormant and only activates when an anomaly is detected during the analysis of compressed data, thereby saving power consumption. The authors of [68] also introduced an event-driven neuromorphic processor which employs a level crossing (LC) sampling method. The LC-ADC captures the level-crossing event and encodes it as a single bit. This approach not only simplifies the ADC quantization process but also minimizes the number of sampling points, thereby reducing the overall data volume.

The event-driven approach must be designed carefully to minimize resource overheads and reduce the latency associated with activating components. Additionally, it is crucial to design the event controller to minimize false positive events, which can lead to unnecessary module activation. Conversely, failing to detect true events can compromise the reliability of the physiological monitoring system.

5.3.2. Duty Cycle Optimization

In contrast to the event-driven approach, where the activation of a component is based on specific events rather than a time period, the duty cycling approach involves switching between active and idle modes at predetermined intervals [118]. This approach effectively manages average power consumption by balancing active and idle periods in systems which can tolerate a wake-up delay. Respecting these conditions is critical; otherwise, the system might miss crucial events. Duty cycling approaches contribute to reducing average power consumption by allocating more time to sleep for energy-consuming components such as wireless communication modules, processing elements (PEs), and sensors.

The author of [119] used the duty cycle optimization approach to reduce the power consumption of high LED currents in the acquisition of PPG signals. The red and IR LEDs were switched on and off alternatively while controlling the duty cycle through digital feedback using pulse density modulation (PDM). This approach avoids unnecessary turning on LEDs to avoid power consumption. Likewise, in [105], the authors applied duty cycle optimization in their system, which has two transmission modes for lossy and lossless compressed ECG data. They configured the Bluetooth transceiver in such a way that the connection interval of the transceiver was managed according to the transmission mode by

adjusting the duty cycle. This approach further reduces power consumption when only lossy data are transmitted.

Duty cycling introduces some latency due to the time needed to switch between modes, which should be acceptable in the context of real-time health monitoring. While the accuracy of results might not be directly affected, the timing and synchronization of data collection could be impacted. This could potentially lead to missed data points if not managed correctly. However, this impact can be minimized by ensuring the wake-up intervals are short enough to capture the necessary data.

5.3.3. TDM of Hardware Components

Another approach is to share the hardware components in a time-division multiplexing fashion with other tasks instead of putting them to sleep. By sharing hardware resources among multiple tasks in a time-sliced manner, the utilization efficiency of those resources can be improved. This leads to reduced energy usage, as it avoids the need to power multiple components simultaneously when a single one can be time-shared.

The authors of [67] classified ECG signals by converting them first into a binary image using a quantized multi-layer perceptron (qMLP) and then classifying them using a binary convolutional neural network (bCNN) to reduce resource and power consumption. The authors orchestrated their algorithm such that IP MACs (multiply–accumulate) were shared between the three dense layers of the qMLP, which not only reduced the resource consumption but also enabled authors to implement their design on the ultra-low power Lattice iCE40 FPGA, which resulted in an overall power reduction.

Although this approach reduces power and resource consumption, it can impact latency due to time-sharing. This means there may be delays in task execution as each task waits for its turn to access the hardware. Therefore, this approach is suitable for scenarios where strict real-time monitoring is not required, such as heart rate monitoring.

The benefits and associated constraints of these optimization techniques are summarized in Table 5 along with their descriptions.

Table 5. Optimization strategies for energy-efficient task scheduling.

Strategy	Description	Advantages	Constraints
Event-driven approach	Activation of hardware components only in response to specific events, rather than running them continuously.	Avoid wasteful energy expenditure during idle periods.	Introduces latency associated with activation time and sharing of resources. Could compromise reliability if a true event is missed.
Duty cycle optimization	Optimizes average power consumption by balancing active and idle periods at predetermined intervals.	Balance the on and off periods to optimize power consumption. Minimize switching activities.	
TDM of hardware components	Share the hardware components among multiple tasks in a time-sliced manner.	Reduce energy usage by avoiding powering up multiple components simultaneously when a single one can be time-shared. Efficient resource management.	

5.4. Exploiting FPGA Features

FPGAs provide unique opportunities for power optimization due to their reconfigurability, customizability, and fine-grained control. These features allow designers to tailor FPGAs precisely to their application's needs, resulting in energy-efficient designs. They offer an opportunity to optimize specific parts of a design by leveraging hardware capabilities such as clock gating, power gating, and dynamic partial reconfiguration (DPR) [120]. These gating techniques not only enable the selective activation or deactivation of components but also facilitate the design of fine-grained power modes. On the other hand, DPR allows for the time-shared deployment of hardware components, further enhancing resource and energy efficiency.

5.4.1. Dynamic Partial Reconfiguration

Partial reconfiguration (PR) allows for modifying parts of the FPGA while other sections continue to operate, leading to significant energy savings. By only utilizing resources when needed, PR minimizes idle power consumption and reduces the necessity to power the entire FPGA continuously. Similarly, the dynamic partial reconfiguration (DPR) of FPGAs allows them to adjust to different tasks or conditions on the fly, saving energy by updating existing resources to meet new requirements and avoiding unnecessary parallel deployments of all modules [121]. Consequently, PR and DPR not only enhance the flexibility and efficiency of FPGA usage but also contribute to the overall energy efficiency.

For instance, a reconfigurable approach was applied in [80] to design an epileptic seizure detection system based on multiple feature extractors followed by one classifier. These feature extractors were implemented using DPR to avoid leakage power from different extractor parts when they were inactive. This resulted in a 0.68× reduction in resource consumption, thereby reducing associated leakage power. Since leakage power is the energy consumed by a component even when it is inactive, delaying the deployment of that component until it is required can eliminate the associated leakage power of that component.

In another instance, the authors of [83], in arrhythmia classification using the DCNN model, conducted convolutional operation layer by layer using a reconfigurable convolution accelerator through time-division multiplexing. This approach not only allowed for the optimized use of resources but also a reduction in idle power consumption by those unused accelerators.

5.4.2. Custom Clock Gating

FPGAs are highly customizable, allowing designers to use clock-gating strategies for a specific application at the fine-grained level. By selectively enabling clocks of different modules only when needed, custom clock gating can minimize significant power consumption in unnecessary clock transitions. The authors of [87] implemented their design using a modular approach for P-wave, T-wave, and QRS extractors. They incorporated the method of custom clock gating to turn off the modules which were not required to be active. Likewise, the authors of [81] also used clock gating to activate only 1 of the 15 multiplier instances at a time, reducing dynamic power consumption by exploiting the fact that these multipliers were used serially.

Clock gating reduces unnecessary switching activities, which in turn lowers dynamic power consumption. Dynamic power consumption mainly arises from the charging and discharging of capacitive loads during clock transitions. By minimizing the number of active clock signals, these transitions are reduced, leading to lower power consumption. Thus, selectively enabling or disabling the clock signal to different parts of the circuit effectively reduces dynamic power consumption. This approach is highly useful for designing various event-driven and sleep mode strategies for applications to optimize power consumption.

5.4.3. Fine-Grained Power Modes

FPGAs allow granular control over power modes for shutting down any module not required for a given task. Such control over power modes reduces not only dynamic power consumption but also static power consumption. FPGAs consume static power even when idle due to transistor leakage [122]. Fine-grained power modes minimize this static power by selectively gating unused components. A similar approach was used in [123] to put a complex module in sleep mode until required. From analysis of the patient-specific seizure patterns, the authors noticed biases in distribution over time. Therefore, they created two different, simple, and complex models for seizure detection. During the time of day when the likelihood of occurrence of seizures is low, a simple model can be activated to lower the power consumption. Since the likelihood of occurrence over the time of a day is extremely low, the system will mostly remain in low power mode, and much power can be saved.

Implementing power down modes for inactive components can result in significant energy savings. For instance, placing communication interfaces, unused sensors, or an associated ADC [124] into sleep mode can reduce the system's power consumption, thereby extending battery life in WMDs. However, there are various levels of power down modes, such as low power, ultra-low power, sleep mode, and deep sleep mode. It is crucial to consider the wake-up time of these modes to prevent missing critical events in continuous health monitoring.

The advantages and constraints associated with these optimization strategies are summarized in Table 6 along with their descriptions.

Table 6. Optimization strategies while exploiting FPGA features.

Strategy	Description	Advantages	Constraints
Dynamic partial reconfiguration	A distinct feature of FPGAs used to reconfigure parts of the FPGA on the fly for resource optimization. Used in event-driven & TDM approaches.	Reduce both dynamic and static power consumption.	Introduces latency associated with activation or wake-up time. Could compromise the reliability if a true event is missed.
Custom clock gating	Selective enabling of the clocks of different modules to reduce unnecessary switching activities. Enables event-driven, duty-cycling and fine-grained power modes.	Minimize switching activities. Reduces dynamic power consumption.	
Fine-grained power modes	Putting the components to sleep or into deep sleep mode when in an idle state.	Reduce both dynamic and static power consumption.	

6. Analyses and Discussion

The increasing demand for real-time physiological monitoring and feedback has significantly driven the incorporation of AI techniques in wearable medical devices. At the same time, there is a rising trend toward integrating multimodal sensors within these devices to achieve comprehensive monitoring of vital parameters, which has increased the data volumes to be stored, transmitted, and processed. Additionally, concerns regarding privacy, security, and the load on cloud computing resources have prompted a shift toward edge computing, where data are processed locally on a device. These advancements have collectively led to a substantial increase in data volumes, computational complexity, memory accesses, and transmission load, which are significant challenges in the design of wearable medical devices. Moreover, the continuous evolution of AI techniques and frequent customization of WMD features for individual patients demand flexibility in reconfigurability, adaptability, and scalability. To address the demand for additional computational resources and flexibility, researchers are exploiting FPGAs due to their unique properties of parallelism, reconfigurability, and partial reconfigurability. Although FPGAs are more power-efficient compared with GPUs and GPPs, they are still power-hungry devices. Consequently, researchers have applied various energy optimization strategies to make FPGAs viable for use in battery-powered WMDs.

Designing an Energy-Conscious Algorithm: It can be observed that most authors tried to design *energy-conscious algorithms* for optimized implementation of AI algorithms on resource-constrained devices while maintaining an acceptable level of accuracy. Since full-scale trained AI models are usually large, optimizations such as *simplification of arithmetic operations*, *parameter quantization*, *feature reduction*, and *signal format conversion* are used to reduce resource consumption. The primary focus of these optimizations is to achieve a trade-off between resource usage and accuracy for better performance, with power optimization as a secondary benefit. FPGAs, being more flexible due to their reconfigurable nature for custom logic design, allow designers to choose any bit width for their operations. Customizing the bit width (e.g., in parameter quantization) can simplify computations, reduce the number of required logic gates, and optimize resource usage, thereby reducing power consumption. Moreover, the parallel processing capabilities of FPGAs are well suited for algorithmic optimization. Unlike sequential processors, FPGAs

can execute multiple operations concurrently, leading to faster processing times and, consequently, less energy consumption for a given task. Optimizing algorithms to leverage this parallelism can result in significant power savings, as the FPGA can complete tasks more efficiently than other platforms which might require higher clock speeds and more power to achieve the same performance. However, careful design of such optimization strategies is essential, as they can impact the accuracy and precision of results, which are critical in physiological monitoring.

Minimizing the Transmission Energy Cost: Since wireless data transmission contributes significantly to the power consumption of WMDs [20], researchers have employed different data reduction techniques, including *data compression*, *compressed sensing*, and *compressed learning*, along with *power-aware transmission strategies*. However, traditional compression methods necessitate a reconstruction step at the receiver, incurring additional computational costs and power usage. To handle this, some studies explored a *compressed learning* approach, where data acquired through sub-Nyquist sampling are directly processed. The effectiveness of this approach relies on the sparsity of the signal. FPGAs offer the opportunity to minimize the latency introduced by compression and reconstruction processes due to their capability of parallel execution. Additionally, the integration of AI with digital signal processing for compressed learning can also benefit from this flexibility of FPGAs. Moreover, this parallelism opportunity is also well suited for the compression of multimodal signals simultaneously [109]. However, while compression techniques offer significant advantages in terms of data volume reduction and power consumption, they also introduce challenges which need to be carefully managed to ensure the accuracy and reliability of physiological monitoring in wearable medical devices.

Energy-Efficient Task Scheduling: Tasks scheduling is another area where we observed an increased focus of researchers to reduce idle power consumption. The designers exploited the event-based nature of underlying applications to schedule tasks using either event triggers or periodic activation. Event-driven design is a reactive approach where a component is activated based on the occurrence of an event. In contrast, duty cycling is proactive, with components activated according to a preset schedule. On the other hand, the time-division multiplexing approach for hardware components leverages relaxed sampling deadlines in physiological monitoring to reduce power consumption by avoiding the simultaneous activation of multiple components. The main advantage of these scheduling approaches is that even though they introduce some latency associated with the activation time, they do not directly affect the accuracy of the results. This is highly important in physiological monitoring. Usually, the sampling deadlines in physiological monitoring are flexible [19], and thus a small delay could be acceptable. FPGAs' unique features of reconfigurability, customizability, and fine-grained control enable designers to schedule tasks more flexibly and at finer levels, optimizing both active and idle periods more effectively. These approaches significantly reduce the leakage and idle power consumption of components which are not active, along with dynamic power consumption by reducing unnecessary switching activities. However, it is crucial to consider the activation time when designing these scheduling approaches to avoid missing any critical events due to activation delays.

Exploiting FPGA Features: On the other hand, rather few works have exploited FPGA features for power reduction, such as dynamic partial reconfiguration and custom clock gating. FPGAs allow the optimization of different design components through fine-grained power modes using custom clock gating and power gating. Specifically, when addressing the lenient sampling deadlines associated with physiological monitoring, there is an opportunity to leverage fine-grained power modes and custom clock-gating techniques. On the other hand, dynamic partial reconfiguration allows for the selective deployment of hardware components only when needed, leading to efficient resource utilization and reduced static power consumption. DPR is highly beneficial in scenarios where the algorithm can be broken down into sequential, functional blocks using a modular approach. In these scenarios, the output of the first block triggers the activation of subsequent blocks.

This allows DPR to delay deployment until necessary, reducing static power consumption. Such deployments, based on fine-grained power modes and DPR, may also introduce some latency due to the gating-up or deployment time. However, they do not directly affect the accuracy of the results, which is extremely crucial in healthcare.

Development Complexities and Financial Barriers: Despite the need for power optimization, developers also face several other difficulties, particularly regarding tools, programming languages, licenses, and costs. Development tools like Xilinx Vivado and Intel Quartus are complex and have steep learning curves, making them difficult to master. Debugging FPGA designs is also more challenging compared with software debugging due to the less intuitive tools. Programming FPGAs typically involves using hardware description languages (HDLs) like VHDL or Verilog, which require a different mindset focused on hardware parallelism and timing. While high-level synthesis (HLS) tools allow for programming in higher-level languages like C or C++, they are not as mature and can produce less efficient designs if proper design space exploration for suitable optimization directives is not performed, which is also time-consuming. Additionally, the cost of licenses for FPGA development tools can be prohibitive, especially for small companies or individual developers. The reliance on vendor-specific tools and IP cores can lead to vendor lock-in, limiting flexibility and increasing costs. Moreover, FPGAs remain expensive devices due to their high degree of flexibility and relatively limited production volumes compared with microcontrollers.

Shortcomings and Future Research Directions: It has been observed that most researchers targeted high-end FPGAs, which inherently have higher power consumption. Ultra-low power flash-based FPGAs, along with hybrid model FPGAs, could be explored for power-sensitive WMDs. Moreover, some fundamental features of FPGAs, such as reconfigurability, customizability, and fine-grained control, are underutilized. The partial reconfiguration capability of FPGA helps save power by delaying the implementation of a specific part until it is needed during operation [80]. Specifically, when addressing the relaxed sampling deadlines associated with physiological monitoring [19], there is an opportunity to leverage *fine-grained power modes* and *custom clock-gating* techniques. These features could be useful when adopting a modular approach to achieve power reduction:

- *Flash-based and hybrid FPGAs*, which are designed for low-power applications, should be further explored for power-sensitive WMDs.
- *Dynamic partial reconfiguration* should also be further exploited in scenarios where the algorithm can be decomposed into sequential functional blocks. In such cases, the output of the first block determines the activation of the subsequent block, a characteristic inherently present in physiological monitoring. By leveraging DPR, the implementation of a subsequent block can be delayed until it is needed to avoid static leakage.
- The capability of *power gating* was also found to be underutilized in the reviewed literature related to WMDs for physiological monitoring. This can also be explored in conjunction with *fine-grained power modes* to significantly reduce static power consumption. This approach works by cutting power to inactive logic blocks, thereby minimizing leakage currents. This approach can introduce latency but does not directly impact the accuracy of the results.
- Exploring external *voltage scaling* for physiological monitoring is also essential. Dynamic power consumption (P_{dyn}) is directly proportional to the square of the operating voltage (V) [125], as described by the equation $P_{dyn} = \alpha C V^2 f$, where C is the switching capacitance, f is the clock frequency, and α is a constant representing the switching factor [126]. This squared relationship between P_{dyn} and V can be exploited to significantly lower power usage, provided that the reliability of operation remains intact, as lowering the voltage increases the switching time and reduces the maximum operating frequency.
- Moreover, both the voltage and frequency can be dynamically adjusted based on performance requirements through *dynamic voltage and frequency scaling (DVFS)*. This

technique can also be explored in the context of physiological monitoring by using WMDs for power optimization while maintaining the necessary performance [126].

- Despite its benefits, *compressed learning* was found to be underutilized in the reviewed articles, requiring further investigation by exploring the sparse nature of physiological signals.

In conclusion, FPGAs offer significant power optimization potential due to their reconfigurability, customizability, and granular control. Additionally, the flash-based and hybrid FPGAs, which are designed for low-power applications, could be used for power-sensitive WMDs. Moreover, features like clock gating, power gating, and dynamic partial reconfiguration enable designers to create energy-efficient designs tailored to specific application needs. For instance, *compressive sensing* can minimize sampling activity at the sensing stage, complemented by an *event-driven* approach using *clock gating* to activate sensors only when necessary. At the computing stage, a *compressed learning* approach can be used for analytics, avoiding reconstruction overheads. The algorithm can be modularized and deployed in a *time-division multiplexed* fashion by using a *partial reconfiguration* to reduce idle power consumption. This comprehensive strategy reduces the unnecessary sensor on time, minimizes redundant data sampling, decreases transmission and computation loads, and reduces power leakage, significantly lowering energy consumption.

7. Conclusions

This work qualitatively reviewed state-of-the-art power optimization strategies in FPGA-based wearable medical devices for physiological monitoring. The benefits and constraints of these techniques were summarized, along with their effects on power, performance, and resource consumption. Potential shortcomings, challenges, and solutions were highlighted, providing guidelines and future research directions for WMD designers.

Our findings indicate that power optimization strategies often optimize resource consumption at the cost of performance or accuracy. Algorithmic optimizations and transmission load reductions decrease resource and power consumption, potentially improving latency but compromising result accuracy. In contrast, task scheduling approaches combined with dynamic partial reconfiguration and clock and power gating reduce power consumption by delaying component activation until necessary. This approach affects latency but maintains accuracy, which is crucial for physiological monitoring. FPGAs offer significant potential for such deployments due to their customizability and reconfigurability, helping reduce idle power consumption without compromising accuracy. Moreover, the flash and hybrid-based FPGAs, designed for low-power applications, are under-explored.

Looking forward, several promising areas for future research have been identified. Power-gating, voltage-scaling, and sleep mode techniques remain underutilized in WMDs and present opportunities for more energy-efficient implementations in physiological monitoring devices. Enhancing the capabilities of power gating and voltage scaling at fine-grained levels through FPGA vendors could offer more flexible options. Partial reconfiguration should be further explored in scenarios where algorithms can be decomposed into sequential functional blocks to minimize static leakage. Additionally, the compressed learning approach is underutilized and could be leveraged by exploiting the sparse nature of physiological signals, optimizing power consumption and resource utilization. Furthermore, flash and hybrid-based FPGAs, designed for low-power applications, could be investigated further for power-sensitive WMDs. By addressing these areas, future research can significantly advance the development of more efficient and effective wearable medical devices.

Author Contributions: Conceptualization, B.d.S.; methodology, M.I.K.; databases search, M.I.K.; analysis, M.I.K.; draft preparation, M.I.K.; review, B.d.S.; editing, M.I.K. and B.d.S.; supervision, B.d.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported through a Ph.D. scholarship from Pakistan and Vrije Universiteit Brussel in Belgium.

Data Availability Statement: This is a qualitative literature review paper. The outcomes of the analysis, which synthesize and analyze the existing state-of-the-art literature, are presented within this paper. No other data were used or generated.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

WMDs	Wearable medical devices
CHM	Continuous health monitoring
LPTs	Low-power techniques
AI	Artificial intelligence
GPP	General purpose processor
GPU	Graphic processing unit
DSP	Digital signal processor
MCU	Microcontroller
FPGAs	Field-programmable gate arrays
ASIC	Application-specific integrated circuits
SRAM	Static random-access memory
IP	Intellectual property
HW	Hardware
SW	Software
SoC	System-on-chip
IoT	Internet of Things
IoWT	Internet of Wearable Things
ML	Machine learning
DL	Deep learning
CNN	Convolutional neural network
WSN	Wireless sensor network
HCR	Human context recognition
bCNN	Binary convolutional neural network
PVC	Premature ventricular contraction
SNN	Spiking neural network
DCNN	Deep convolutional neural network
TNN	Ternary neural network
qMLP	Quantized multi-layer perceptron
DPR	Dynamic partial reconfiguration
CL	Compressed learning
CS	Compressed sensing
TDM	Time-division multiplexing
MAC	Multiply–accumulate
SAC	Shift–accumulate
INT	Integer
FP	Floating point

References

1. Caizzone, A.; Boukhayma, A.; Enz, C. A 2.6 µW Monolithic CMOS Photoplethysmographic (PPG) Sensor Operating with 2 µW LED Power for Continuous Health Monitoring. *IEEE Trans. Biomed. Circuits Syst.* **2019**, *13*, 1243–1253. [CrossRef]
2. Hernandez, N.; Castro, L.; Medina-Quero, J.; Favela, J.; Michán, L.; Mortenson, W.B. Scoping review of healthcare literature on mobile, wearable, and textile sensing technology for continuous monitoring. *J. Healthc. Informatics Res.* **2021**, *5*, 270–299. [CrossRef]
3. Nia, A.M.; Mozaffari-Kermani, M.; Sur-Kolay, S.; Raghunathan, A.; Jha, N.K. Energy-Efficient Long-term Continuous Personal Health Monitoring. *IEEE Trans. Multi-Scale Comput. Syst.* **2015**, *1*, 85–98. [CrossRef]
4. Liang, Y.; Zhao, C.Z.; Yuan, H.; Chen, Y.; Zhang, W.; Huang, J.Q.; Yu, D.; Liu, Y.; Titirici, M.M.; Chueh, Y.L.; et al. A review of rechargeable batteries for portable electronic devices. *InfoMat* **2019**, *1*, 6–32. [CrossRef]
5. EU. Next Generation Power Sources for Self-sustainable Devices—Integrated Multi-source Energy Harvesters. 2023. Available online: https://cordis.europa.eu/project/id/705437 (accessed on 4 March 2024).

6. Singh, P.; Pandey, B.; Bhandari, N.; Bisht, S.; Bisht, N.; Budhani, S.K. Design of Energy Efficient IoMT Electrocardiogram (ECG) Machine on 28 nm FPGA. In *Towards the Integration of IoT, Cloud and Big Data: Services, Applications and Standards*; Rishiwal, V., Kumar, P., Tomar, A., Malarvizhi Kumar, P., Eds.; Springer Nature: Singapore, 2023; pp. 43–55. [CrossRef]
7. Shumba, A.T.; Montanaro, T.; Sergi, I.; Bramanti, A.; Ciccarelli, M.; Rispoli, A.; Carrizzo, A.; De Vittorio, M.; Patrono, L. Wearable Technologies and AI at the Far Edge for Chronic Heart Failure Prevention and Management: A Systematic Review and Prospects. *Sensors* **2023**, *23*, 6896. [CrossRef]
8. Su, X.; An, L.; Cheng, Z.; Weng, Y. Cloud–edge collaboration-based bi-level optimal scheduling for intelligent healthcare systems. *Future Gener. Comput. Syst.* **2023**, *141*, 28–39. [CrossRef]
9. Covi, E.; Donati, E.; Liang, X.; Kappel, D.; Heidari, H.; Payvand, M.; Wang, W. Adaptive extreme edge computing for wearable devices. *Front. Neurosci.* **2021**, *15*, 611300. [CrossRef]
10. Fernández-Caramés, T.M.; Fraga-Lamas, P. Towards The Internet of Smart Clothing: A Review on IoT Wearables and Garments for Creating Intelligent Connected E-Textiles. *Electronics* **2018**, *7*, 405. [CrossRef]
11. Maheepala, M.; Joordens, M.A.; Kouzani, A.Z. Low Power Processors and Image Sensors for Vision-Based IoT Devices: A Review. *IEEE Sensors J.* **2021**, *21*, 1172–1186. [CrossRef]
12. Actel. The Many Flavors of Low-Power, Low-Cost FPGAs. Available online: https://ww1.microchip.com/downloads/aemDocuments/documents/FPGA/ProductDocuments/SupportingCollateral/low_power_wp.pdf (accessed on 25 September 2024).
13. Chéour, R.; Khriji, S.; Houssaini, D.E.; Baklouti, M.; Abid, M.; Kanoun, O. Recent Trends of FPGA Used for Low-Power Wireless Sensor Network. *IEEE Aerosp. Electron. Syst. Mag.* **2019**, *34*, 28–38. [CrossRef]
14. Dattani, S.; Spooner, F.; Ritchie, H.; Roser, M. Causes of Death. Our World in Data. 2023. Available online: https://ourworldindata.org/causes-of-death (accessed on 10 August 2024).
15. WHO. Noncommunicable Diseases. 2023. Available online: https://www.who.int/news-room/fact-sheets/detail/noncommunicable-diseases (accessed on 10 August 2024).
16. Lal, B.; Gravina, R.; Spagnolo, F.; Corsonello, P. Compressed Sensing Approach for Physiological Signals: A Review. *IEEE Sensors J.* **2023**, *23*, 5513–5534. [CrossRef]
17. Chen, C.; Shu, M.; Zhou, S.; Liu, Z.; Liu, R. Wavelet-domain group-sparse denoising method for ECG signals. *Biomed. Signal Process. Control* **2023**, *83*, 104702. [CrossRef]
18. Liu, B.; Zhang, Z.; Xu, G.; Fan, H.; Fu, Q. Energy efficient telemonitoring of physiological signals via compressed sensing: A fast algorithm and power consumption evaluation. *Biomed. Signal Process. Control* **2014**, *11*, 80–88. [CrossRef]
19. Philipp, F.; Glesner, M. A reconfigurable wireless platform for biomedical signal processing. In Proceedings of the 6th 2013 Biomedical Engineering International Conference, Amphur Muang, Thailand, 23–25 October 2013; pp. 1–5. [CrossRef]
20. Wu, D.; Zhao, S.; Yang, J.; Sawan, M. Software-Hardware Co-Design for Energy-Efficient Continuous Health Monitoring via Task-Aware Compression. *IEEE Trans. Biomed. Circuits Syst.* **2023**, *17*, 180–191. [CrossRef]
21. Zeidman, B. All About FPGAs. Available online: https://www.eetimes.com/all-about-fpgas/ (accessed on 25 September 2024).
22. Goldsmith, T. FPGA Basics: LUTs, CLBs, Slices, and Logic Cells. Available online: https://thebitbuilder.com/fpga-basics-luts-clbs-slices-and-logic-cells/ (accessed on 25 September 2024).
23. AMD. Xilinx 7-Series FPGAs Data Sheet. Available online: https://docs.amd.com/v/u/en-US/ds180_7Series_Overview (accessed on 25 September 2024).
24. AMD. Spartan-7 FPGAs' DC and AC Switching Characteristics. Available online: https://docs.amd.com/r/en-US/ds189-spartan-7-data-sheet/DC-Characteristics (accessed on 25 September 2024).
25. AMD. Artix-7 FPGAs' DC and AC Switching Characteristics. Available online: https://docs.amd.com/v/u/en-US/ds181_Artix_7_Data_Sheet (accessed on 25 September 2024).
26. QuickLogic. PolarPro-3 Device Data Sheet. Available online: https://www.quicklogic.com/wp-content/uploads/2022/01/QL_PolarPro_3_Device_Data_Sheet_Oct-13_2021.pdf (accessed on 25 September 2024).
27. QuickLogic. PolarPro-3 Tiny, Low Cost SRAM Reprogrammable FPGAs with Open Source Tools. Available online: https://www.quicklogic.com/products/fpga/fpgas-sram/ (accessed on 25 September 2024).
28. Intel. Cyclone-V Device Overview. Available online: https://www.intel.com/content/www/us/en/docs/programmable/683694/current/maximum-resources-04257.html (accessed on 25 September 2024).
29. Intel. MAX-10 FPGA Device Overview. Available online: https://www.intel.com/content/www/us/en/docs/programmable/683658/current/fpga-device-overview.html (accessed on 25 September 2024).
30. Intel. MAX-10 Power Management User Guide. Available online: https://www.intel.com/content/www/us/en/docs/programmable/683400/18-0/power-supply-design.html (accessed on 25 September 2024).
31. Lattice-Semiconductor. iCE40 UltraPlus Low Power FPGA. Available online: https://www.latticesemi.com/en/Products/FPGAandCPLD/iCE40UltraPlus (accessed on 25 September 2024).
32. Lattice-Semiconductor. iCE40 LP/HX Low-Power, High-Performance FPGA. Available online: https://www.latticesemi.com/Products/FPGAandCPLD/iCE40 (accessed on 25 September 2024).
33. Lattice-Semiconductor. MachXO2 - Bridging and I/O expansion versatility. Available online: https://www.latticesemi.com/Products/FPGAandCPLD/MachXO2 (accessed on 25 September 2024).
34. Microchip-Technology. IGLOO-2 FPGA Product Brief. Available online: https://www.microchip.com/content/dam/mchp/documents/FPGA/ProductDocuments/ProductBrief/PB0121 (accessed on 25 September 2024).

35. Microchip-Technology. IGLOO-2 FPGA. Available online: https://www.microchip.com/en-us/products/fpgas-and-plds/fpgas/igloo-2-fpgas (accessed on 25 September 2024).
36. Microchip-Technology. IGLOO-2 FPGA and SmartFusion2 SoC FPGA. Available online: https://www.digikey.be/en/htmldatasheets/production/2520949/0/0/1/m2gl010-fgg484i#pf17 (accessed on 25 September 2024).
37. Microchip-Technology. IGLOO Nano Low Power Flash FPGAswith Flash-Freeze Technology. Available online: https://ww1.microchip.com/downloads/aemDocuments/documents/FPGA/ProductDocuments/DataSheets/microsemi_ds0110_igloo_nano_low_power_flash_fpgas_ds.pdf (accessed on 25 September 2024).
38. Microchip-Technology. IGLOO Nano FPGAs. Available online: https://www.microchip.com/en-us/products/fpgas-and-plds/fpgas/igloo-fpgas#IGLOO (accessed on 25 September 2024).
39. GOWIN. GW1NZ Series of FPGA Products. Available online: https:///cdn.gowinsemi.com.cn/DS841E.pdf (accessed on 25 September 2024).
40. Intel. PowerPlay Early Power Estimator (EPE) Tool. Available online: https://www.intel.com/content/www/us/en/support/programmable/support-resources/power/max-10-estimator-download.html (accessed on 25 September 2024).
41. Tesema, W.; Jimma, W.; Khan, M.I.; Stiens, J.; da Silva, B. A Taxonomy of Low-Power Techniques in Wearable Medical Devices for Healthcare Applications. *Electronics* **2024**, *13*, 3097. [CrossRef]
42. Ray, P.P.; Dash, D. A Review on Internet of Wearable Things for Pervasive E-Health Care: Energy Efficiency and Prospects. In *Proceedings of the International Conference on Cognitive and Intelligent Computing (ICCIC-2021), Hyderabad, India, 11–12 December 2021*; Kumar, A., Ghinea, G., Merugu, S., Hashimoto, T., Eds.; Springer: Singapore, 2023; pp. 69–82. [CrossRef]
43. Tesema, W.; Da Silva, B.; Jimma, W.; Stiens, J. Power Saving Techniques for Wearable Devices in Medical Applications. In Proceedings of the IECON 2022—48th Annual Conference of the IEEE Industrial Electronics Society, Brussels, Belgium, 17–20 October 2022; pp. 1–8. [CrossRef]
44. Hartmann, M.; Hashmi, U.S.; Imran, A. Edge computing in smart health care systems: Review, challenges, and research directions. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e3710. [CrossRef]
45. Qaim, W.B.; Ometov, A.; Molinaro, A.; Lener, I.; Campolo, C.; Lohan, E.S.; Nurmi, J. Towards Energy Efficiency in the Internet of Wearable Things: A Systematic Review. *IEEE Access* **2020**, *8*, 175412–175435. [CrossRef]
46. Cong, P.; Zhou, J.; Li, L.; Cao, K.; Wei, T.; Li, K. A survey of hierarchical energy optimization for mobile edge computing: A perspective from end devices to the cloud. *ACM Comput. Surv.* **2020**, *53*, 38. [CrossRef]
47. Seneviratne, S.; Hu, Y.; Nguyen, T.; Lan, G.; Khalifa, S.; Thilakarathna, K.; Hassan, M.; Seneviratne, A. A Survey of Wearable Devices and Challenges. *IEEE Commun. Surv. Tutorials* **2017**, *19*, 2573–2620. [CrossRef]
48. Wang, C.; Lu, W.; Narayanan, M.R.; Redmond, S.J.; Lovell, N.H. Low-power technologies for wearable telecare and telehealth systems: A review. *Biomed. Eng. Lett.* **2015**, *5*, 1–9. [CrossRef]
49. Rault, T.; Bouabdallah, A.; Challal, Y.; Marin, F. A survey of energy-efficient context recognition systems using wearable sensors for healthcare applications. *Pervasive Mob. Comput.* **2017**, *37*, 23–44. [CrossRef]
50. Altman, M.B.; Wan, W.; Hosseini, A.S.; Arabi Nowdeh, S.; Alizadeh, M. Machine learning algorithms for FPGA Implementation in biomedical engineering applications: A review. *Heliyon* **2024**, *10*, e26652. [CrossRef]
51. Vaithianathan, M.; Patil, M.; Ng, S.F.; Udkar, S. Energy-Efficient FPGA Design for Wearable and Implantable Devices. *ESP Int. J. Adv. Sci. Technol.* **2024**, *2*, 37–51.
52. Jegan, R.; Nimi, W.S. On the development of low power wearable devices for assessment of physiological vital parameters: A systematic review. *J. Public Health* **2024**, *32*, 1093–1108. [CrossRef]
53. Umapathy, K.; Muthukumaran, D.; Chandramohan, S.; Sivakumar, M.; James, O., Low Power Methodologies for FPGA—An Overview. In *Low Power Architectures for IoT Applications*; Sharma, D.K., Sharma, R., Jeon, G., Polkowski, Z., Eds.; Springer Nature: Singapore, 2023; pp. 85–97. [CrossRef]
54. Ibro, M.; Marinova, G. Review on Low-Power Consumption Techniques for FPGA-based designs in IoT technology. In Proceedings of the 2021 16th International Conference on Telecommunications (ConTEL), Zagreb, Croatia, 30 June–2 July 2021; pp. 110–114. [CrossRef]
55. Jiang, C.; Fan, T.; Gao, H.; Shi, W.; Liu, L.; Cérin, C.; Wan, J. Energy aware edge computing: A survey. *Comput. Commun.* **2020**, *151*, 556–580. [CrossRef]
56. Mittal, S. A survey of FPGA-based accelerators for convolutional neural networks. *Neural Comput. Appl.* **2020**, *32*, 1109–1139. [CrossRef]
57. Blaiech, A.G.; Ben Khalifa, K.; Valderrama, C.; Fernandes, M.A.; Bedoui, M.H. A Survey and Taxonomy of FPGA-based Deep Learning Accelerators. *J. Syst. Archit.* **2019**, *98*, 331–345. [CrossRef]
58. Abdullah, W.A.N.W.; Yaakob, N.; Elobaid, M.E.; Warip, M.N.M.; Yah, S.A. Energy-efficient remote healthcare monitoring using IoT: A review of trends and challenges. In Proceedings of the ICC '16: International Conference on Internet of Things and Cloud Computing, Cambridge, UK, 22–23 March 2016; pp. 1–8. [CrossRef]
59. Su, M.; Hua, J.; Sun, X.; Liu, Z.; Shi, Y.; Pan, L. Wireless Wearable Devices and Recent Applications in Health Monitoring and Clinical Diagnosis. *Biomed. Mater. Devices* **2024**, *2*, 669–694. [CrossRef]
60. Sun, X.; Zhao, C.; Li, H.; Yu, H.; Zhang, J.; Qiu, H.; Liang, J.; Wu, J.; Su, M.; Shi, Y.; et al. Wearable Near-Field Communication Sensors for Healthcare: Materials, Fabrication and Application. *Micromachines* **2022**, *13*, 784. [CrossRef]

61. Kim, H.; Rigo, B.; Wong, G.; Lee, Y.J.; Yeo, W.H. Advances in wireless, batteryless, implantable electronics for real-time, continuous physiological monitoring. *Nano-Micro Lett.* **2024**, *16*, 52. [CrossRef]
62. Ran, S.C.; Wang, Q.A.; Wang, J.F.; Ni, Y.Q.; Guo, Z.X.; Luo, Y. A Concise State-of-the-Art Review of Crack Monitoring Enabled by RFID Technology. *Appl. Sci.* **2024**, *14*, 3213. [CrossRef]
63. Liu, G.; Wang, Q.A.; Jiao, G.; Dang, P.; Nie, G.; Liu, Z.; Sun, J. Review of wireless RFID strain sensing technology in structural health monitoring. *Sensors* **2023**, *23*, 6925. [CrossRef]
64. Wang, Q.A.; Zhang, C.; Ma, Z.G.; Jiao, G.Y.; Jiang, X.W.; Ni, Y.Q.; Wang, Y.C.; Du, Y.T.; Qu, G.B.; Huang, J. Towards long-transmission-distance and semi-active wireless strain sensing enabled by dual-interrogation-mode RFID technology. *Struct. Control Health Monit.* **2022**, *29*, e3069. [CrossRef]
65. Ben Dhaou, I.; Ebrahimi, M.; Ben Ammar, M.; Bouattour, G.; Kanoun, O. Edge devices for internet of medical things: Technologies, techniques, and implementation. *Electronics* **2021**, *10*, 2104. [CrossRef]
66. Silvestri, F.; Acciarito, S.; Cardarilli, G.C.; Khanal, G.M.; Di Nunzio, L.; Fazzolari, R.; Re, M. FPGA Implementation of a Low-Power QRS Extractor. In *Proceedings of the Applications in Electronics Pervading Industry, Environment and Society (ApplePies 2017), Rome, Italy, 20–21 September 2017*; De Gloria, A., Ed.; Lecture Notes in Electrical Engineering; Springer: Cham, Switzerland, 2019; Volume 512, pp. 9–15. [CrossRef]
67. Wong, D.L.T.; Li, Y.; John, D.; Ho, W.K.; Heng, C.H. Low Complexity Binarized 2D-CNN Classifier for Wearable Edge AI Devices. *IEEE Trans. Biomed. Circuits Syst.* **2022**, *16*, 822–831. [CrossRef]
68. Chu, H.; Yan, Y.; Gan, L.; Jia, H.; Qian, L.; Huan, Y.; Zheng, L.; Zou, Z. A Neuromorphic Processing System with Spike-Driven SNN Processor for Wearable ECG Classification. *IEEE Trans. Biomed. Circuits Syst.* **2022**, *16*, 511–523. [CrossRef]
69. Hosseini, M.; Paneliya, H.; Kallakuri, U.; Khatwani, M.; Mohsenin, T. Minimizing Classification Energy of Binarized Neural Network Inference for Wearable Devices. In Proceedings of the 20th International Symposium on Quality Electronic Design (ISQED), Santa Clara, CA, USA, 6–7 March 2019; pp. 259–264. [CrossRef]
70. Kang, S.; Kim, H.; Park, C.; Sim, Y.; Lee, S.; Jung, Y. sEMG-Based Hand Gesture Recognition Using Binarized Neural Network. *Sensors* **2023**, *23*, 1436. [CrossRef]
71. Hassan, O.; Paul, T.; Amin, N.; Titirsha, T.; Thakker, R.; Parvin, D.; Mosa, A.S.M.; Islam, S.K. An Optimized Hardware Inference of SABiNN: Shift-Accumulate Binarized Neural Network for Sleep Apnea Detection. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 2516311. [CrossRef]
72. Makhlooghpour, A.; Ahmadi, A. A Dual Stage Resource Efficient ECG Classifier. In Proceedings of the 2023 IEEE Biomedical Circuits and Systems Conference (BioCAS), Toronto, ON, Canada, 19–21 October 2023; pp. 1–5. [CrossRef]
73. Ercan, R.; Xia, Y.; Zhao, Y.; Loureiro, R.; Yang, S.; Zhao, H. An Ultralow-Power Real-Time Machine Learning Based fNIRS Motion Artifacts Detection. *IEEE Trans. Very Large Scale Integr. Syst.* **2024**, *32*, 763–773. [CrossRef]
74. Zaman, K.S.; Reaz, M.M.B.I. Secure and efficient implementation of facial emotion detection for smart patient monitoring system. *Quant. Biol.* **2023**, *11*, 175–182. [CrossRef]
75. Aruna, V.B.K.L.; Ekambaram, C.; Padmaja, M. Field programmable gate array implementation of an adaptive filtering based noise reduction and enhanced compression technique for healthcare applications. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4654. [CrossRef]
76. Fan, J.; Yang, S.; Liu, J.; Zhu, Z.; Xiao, J.; Chang, L.; Lin, S.; Zhou, J. A High Accuracy & Ultra-Low Power ECG-Derived Respiration Estimation Processor for Wearable Respiration Monitoring Sensor. *Biosensors* **2022**, *12*, 665. [CrossRef]
77. Hassan, O.; Paul, T.; Shuvo, M.M.H.; Parvin, D.; Thakker, R.; Chen, M.; Mosa, A.S.M.; Islam, S.K. Energy Efficient Deep Learning Inference Embedded on FPGA for Sleep Apnea Detection. *J. Signal Process. Syst.* **2022**, *94*, 609–619. [CrossRef]
78. Gon, A.; Mukherjee, A. Design of hardware-efficient PVC recognition and classification system for early detection of sudden cardiac arrests. *AEU-Int. J. Electron. Commun.* **2023**, *172*, 154955. [CrossRef]
79. Chinmayi, K.; Padmaja, M. VLSI implementation of ECG feature extraction using Integer Haar Wavelet and EMD Algorithm. In Proceedings of the 2023 First International Conference on Cyber Physical Systems, Power Electronics and Electric Vehicles (ICPEEV), Hyderabad, India, 28–30 September 2023; pp. 1–6. [CrossRef]
80. Fawzy, M.; Hussien, A.; Mostafa, H. FPGA Utilized Implementation of Epileptic Seizure Detection System Based on Wearable Devices using Dynamic Partial Reconfiguration. In Proceedings of the 2022 10th International Japan-Africa Conference on Electronics, Communications, and Computations (JAC-ECC), Alexandria, Egypt, 19–20 December 2022; pp. 119–124. [CrossRef]
81. Taufique, Z.; Kanduri, A.; Bin Altaf, M.A.; Liljeberg, P. Approximate Feature Extraction for Low Power Epileptic Seizure Prediction in Wearable Devices. In Proceedings of the 2021 IEEE Nordic Circuits and Systems Conference (NorCAS), Oslo, Norway, 26–27 October 2021; pp. 1–7. [CrossRef]
82. Gonzalez, H.A.; Muzaffar, S.; Yoo, J.; Elfadel, I.A.M. An Inference Hardware Accelerator for EEG-Based Emotion Detection. In Proceedings of the 2020 IEEE International Symposium on Circuits and Systems (ISCAS), Seville, Spain, 12–14 October 2020; pp. 1–5. [CrossRef]
83. Ran, S.; Yang, X.; Liu, M.; Zhang, Y.; Cheng, C.; Zhu, H.; Yuan, Y. Homecare-Oriented ECG Diagnosis With Large-Scale Deep Neural Network for Continuous Monitoring on Embedded Devices. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 2503113. [CrossRef]

84. Hassan, O.; Thakker, R.; Paul, T.; Parvin, D.; Mohammad Mosa, A.S.; Islam, S.K. SABiNN: FPGA Implementation of Shift Accumulate Binary Neural Network Model for Real-Time Automatic Detection of Sleep Apnea. In Proceedings of the 2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Ottawa, ON, Canada, 16–19 May 2022; pp. 1–6. [CrossRef]
85. Hu, J.; Goh, W.L.; Gao, Y. Classification of ECG Anomaly with Dynamically-biased LSTM for Continuous Cardiac Monitoring. In Proceedings of the 2023 IEEE International Symposium on Circuits and Systems (ISCAS), Monterey, CA, USA, 21–25 May 2023; pp. 1–5. [CrossRef]
86. Liu, W.; Guo, Q.; Chen, S.; Chang, S.; Wang, H.; He, J.; Huang, Q. A fully-mapped and energy-efficient FPGA accelerator for dual-function AI-based analysis of ECG. *Front. Physiol.* **2023**, *14*, 1079503. [CrossRef]
87. Janveja, M.; Parmar, R.; Trivedi, G.; Jan, P.; Nemec, Z. An Energy Efficient and Resource Optimal VLSI Architecture for ECG Feature Extraction for Wearable Healthcare Applications. In Proceedings of the 2022 32nd International Conference Radioelektronika (RADIOELEKTRONIKA), Kosice, Slovakia, 21–22 April 2022; pp. 1–6. [CrossRef]
88. Kim, J.K.; Oh, J.H.; Gwon, O.S.; Yang, J.W.; Lee, S.E. Real-time PPG monitoring system for mobile healthcare devices. In Proceedings of the 2017 IEEE International Symposium on Consumer Electronics (ISCE), Kuala Lumpur, Malaysia, 14–15 November 2017; pp. 84–85. [CrossRef]
89. Giorgio, A.; Guaragnella, C.; Rizzi, M. FPGA-Based Decision Support System for ECG Analysis. *J. Low Power Electron. Appl.* **2023**, *13*, 6. [CrossRef]
90. Tian, J.; Mercier, P.; Paolini, C. Ultra low-power, wearable, accelerated shallow-learning fall detection for elderly at-risk persons. *Smart Health* **2024**, *33*, 100498. [CrossRef]
91. Alam, M.S.; Siddiqui, Y.; Hasan, M.; Farooq, O.; Himeur, Y. Energy-Efficient FPGA Based Sleep Apnea Detection Using EEG Signals. *IEEE Access* **2024**, *12*, 40182–40195. [CrossRef]
92. Elbedwehy, A.N.; El-Mohandes, A.M.; Elnakib, A.; Abou-Elsoud, M.E. FPGA-based reservoir computing system for ECG denoising. *Microprocess. Microsyst.* **2022**, *91*, 104549. [CrossRef]
93. Gu, M.; Zhang, Y.; Wen, Y.; Ai, G.; Zhang, H.; Wang, P.; Wang, G. A lightweight convolutional neural network hardware implementation for wearable heart rate anomaly detection. *Comput. Biol. Med.* **2023**, *155*, 106623. [CrossRef]
94. Razi, K.F.; Schmid, A. Epileptic Seizure Detection with Patient-Specific Feature and Channel Selection for Low-power Applications. *IEEE Trans. Biomed. Circuits Syst.* **2022**, *16*, 626–635. [CrossRef]
95. Abubakar, S.M.; Saadeh, W.; Altaf, M.A.B. A wearable long-term single-lead ECG processor for early detection of cardiac arrhythmia. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018; pp. 961–966. [CrossRef]
96. Attaran, N.; Puranik, A.; Brooks, J.; Mohsenin, T. Embedded Low-Power Processor for Personalized Stress Detection. *IEEE Trans. Circuits Syst. II Express Briefs* **2018**, *65*, 2032–2036. [CrossRef]
97. Harpale, V.; Bairagi, V. An adaptive method for feature selection and extraction for classification of epileptic EEG signal in significant states. *J. King Saud Univ.-Comput. Inf. Sci.* **2021**, *33*, 668–676. [CrossRef]
98. Rawal, V.; Prajapati, P.; Darji, A. Hardware implementation of 1D-CNN architecture for ECG arrhythmia classification. *Biomed. Signal Process. Control* **2023**, *85*, 104865. [CrossRef]
99. Indira, P.B.; Krishna, R.D. Optimized adaptive neuro fuzzy inference system (OANFIS) based EEG signal analysis for seizure recognition on FPGA. *Biomed. Signal Process. Control* **2021**, *66*, 102484. [CrossRef]
100. Liu, J.; Qiu, H.; Wang, X.; Qin, H.; Zhou, Y.; Zhou, J. A High Accuracy & Ultra-Low Power PPG-Derived HR Estimation AI Processor for Wearable Devices. In Proceedings of the 2023 6th International Conference on Electronics Technology (ICET), Chengdu, China, 12–15 May 2023; pp. 1103–1107. [CrossRef]
101. Abubakar, S.M.; Yin, Y.; Tan, S.; Jiang, H.; Wang, Z.; U, S.P.; Jia, W. A 2.52 μA Wearable Single Lead Ternary Neural Network Based Cardiac Arrhythmia Detection Processor. In Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 22–28 May 2021; pp. 1–4. [CrossRef]
102. Khan, M.R.; Saadeh, W.; Altaf, M.A.B. A low complexity patient-specific threshold based accelerator for the Grand-mal seizure disorder. In Proceedings of the 2017 IEEE Biomedical Circuits and Systems Conference (BioCAS), Turin, Italy, 19–21 October 2017; pp. 1–4. [CrossRef]
103. Chen, W.J.; Lu, C.C.; Jiang, Y.H.; Fang, H.W. FPGA design and verification for lossless ECG coding compression. In Proceedings of the 2017 6th International Symposium on Next Generation Electronics (ISNE), Keelung, Taiwan, 23–25 May 2017; pp. 1–3. [CrossRef]
104. Kulau, U.; Ahmed, A.N.A. Efficient Online Compression for MEMS based BCG Wearable Sensors on ULP FPGA. In Proceedings of the 2023 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL), Lihue, HI, USA, 28–31 March 2023; pp. 1–4. [CrossRef]
105. Deepu, C.J.; Heng, C.H.; Lian, Y. A Hybrid Data Compression Scheme for Power Reduction in Wireless Sensors for IoT. *IEEE Trans. Biomed. Circuits Syst.* **2017**, *11*, 245–254. [CrossRef]
106. Donoho, D. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
107. Gurve, D.; Delisle-Rodriguez, D.; Bastos-Filho, T.; Krishnan, S. Trends in compressive sensing for EEG signal processing applications. *Sensors* **2020**, *20*, 3703. [CrossRef]

108. Kerdjidj, O.; Amira, A.; Ghanem, K.; Ramzan, N.; Katsigiannis, S.; Chouireb, F. An FPGA implementation of the matching pursuit algorithm for a compressed sensing enabled e-Health monitoring platform. *Microprocess. Microsyst.* **2019**, *67*, 131–139. [CrossRef]
109. Liu, D.; Wang, Q.; Zhang, Y.; Liu, X.; Lu, J.; Sun, J. FPGA-based real-time compressed sensing of multichannel EEG signals for wireless body area networks. *Biomed. Signal Process. Control* **2019**, *49*, 221–230. [CrossRef]
110. Calderbank, R.; Jafarpour, S.; Schapire, R. Compressed learning: Universal sparse dimensionality reduction and learning in the measurement domain. **2009**, *preprint*. Available online: https://api.semanticscholar.org/CorpusID:15610215 (accessed on 16 October 2024).
111. Lal, B.; Li, Q.; Corsonello, P.; Gravina, R. Abnormal ECG Detection in Wearable Devices Using Compressed Learning. In Proceedings of the 2023 International Conference on Networking, Sensing and Control (ICNSC), Marseille, France, 25–27 October 2023; pp. 1–6. [CrossRef]
112. Li, W.; Chu, H.; Huang, B.; Huan, Y.; Zheng, L.; Zou, Z. Enabling on-device classification of ECG with compressed learning for health IoT. *Microelectron. J.* **2021**, *115*, 105188. [CrossRef]
113. Jafari, A.; Page, A.; Sagedy, C.; Smith, E.; Mohsenin, T. A low power seizure detection processor based on direct use of compressively-sensed data and employing a deterministic random matrix. In Proceedings of the 2015 IEEE Biomedical Circuits and Systems Conference (BioCAS), Atlanta, GA, USA, 22–24 October 2015; pp. 1–4. [CrossRef]
114. Priyadarshini, R.; Shaikh, N.; Godi, R.K.; Dhal, P.; Sharma, R.; Perwej, Y. IOT-based power control systems framework for healthcare applications. *Meas. Sensors* **2023**, *25*, 100660. [CrossRef]
115. Ye, Z.; Lu, X.; Wang, S.; Li, B. An 842 nW Wearable Inter-Patient Cardiac Arrhythmia Monitoring Processor with a Feature Engine-Based Artificial Neural Network. In Proceedings of the 2023 IEEE 15th International Conference on ASIC (ASICON), Nanjing, China, 24–27 October 2023; pp. 1–4. [CrossRef]
116. Fang, C.; Shen, Z.; Tian, F.; Yang, J.; Sawan, M. A Compact Online-Learning Spiking Neuromorphic Biosignal Processor. In Proceedings of the 2022 IEEE International Symposium on Circuits and Systems (ISCAS), Austin, TX, USA, 27 May–1 June 2022; pp. 2147–2151. [CrossRef]
117. Razi, K.F.; Schmid, A. Two-stage Hardware-Friendly Epileptic Seizure Detection Method with a Dynamic Feature Selection. In Proceedings of the 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Mexico, 1–5 November 2021; pp. 156–159. [CrossRef]
118. Chandrakasan, A.; Min, R.; Bhardwaj, M.; Cho, S.; Wang, A. Power aware wireless microsensor systems. In Proceedings of the 28th European Solid-State Circuits Conference, Florence, Italy, 24–26 September 2002; pp. 47–54. [CrossRef]
119. Syed, A.; Khan, K.; Ahmad, A.; Asad, M.S.; Saadeh, W. A 1mW Vitals Monitoring System for Asthmatic Patients based on Photoplethysmography. In Proceedings of the 2019 IEEE Biomedical Circuits and Systems Conference (BioCAS), Nara, Japan, 17–19 October 2019; pp. 1–4. [CrossRef]
120. Babu, P.; Parthasarathy, E. Reconfigurable FPGA architectures: A survey and applications. *J. Inst. Eng. Ser. B* **2021**, *102*, 143–156. [CrossRef]
121. Vipin, K.; Fahmy, S.A. FPGA dynamic and partial reconfiguration: A survey of architectures, methods, and applications. *ACM Comput. Surv.* **2018**, *51*, 1–39. [CrossRef]
122. Bsoul, A.A.M.; Wilton, S.J.E. An FPGA architecture supporting dynamically controlled power gating. In Proceedings of the 2010 International Conference on Field-Programmable Technology, Beijing, China, 8–10 December 2010; pp. 1–8. [CrossRef]
123. Varnosfaderani, S.M.; Rahman, R.; Sarhan, N.J.; Alhawari, M. A Self-Aware Power Management Model for Epileptic Seizure Systems Based on Patient-Specific Daily Seizure Pattern. In Proceedings of the 2023 International Conference on Microelectronics (ICM), Abu Dhabi, United Arab Emirates, 17–20 December 2023; pp. 91–95. [CrossRef]
124. Mo, H.; Zhou, C. Hardware Acceleration of PPG Waveform and Heart Rate Detection System. In Proceedings of the 2023 6th International Conference on Electronics Technology (ICET), Chengdu, China, 12–15 May 2023; pp. 974–979. [CrossRef]
125. Bublitz, L.; Rust, J.; Kulau, U. Let's go below: Potential of Undervolting on Low-Power FPGAs. In Proceedings of the 11th International Workshop on Energy Harvesting & Energy-Neutral Sensing Systems, Istanbul, Türkiye, 12 November 2023; pp. 51–57. [CrossRef]
126. Khriji, S.; Chéour, R.; Kanoun, O. Dynamic voltage and frequency scaling and duty-cycling for ultra low-power wireless sensor nodes. *Electronics* **2022**, *11*, 4071. [CrossRef]