

Article

An Ultra-Low-Power Bit-Serial Variable-Accuracy FFT Processor

Yue Lu , Tom Kazmierski

ECS, University of Southampton, Southampton SO17 1BJ, UK; yl15g13@ecs.soton.ac.uk

* Correspondence: tjkc@soton.ac.uk; Tel.: +44-238-059-3520

Abstract: In this paper, a new approach is proposed for designing ultra-low-power FFT (Fast Fourier Transform) system suitable for use in energy harvesting powered sensors. Bit-serial architecture is adopted to reduce the power consumption of butterfly operation. Simulation results show that, compared with state-of-the-art bit-serial and conventional parallel processors, the proposed technique is superior in terms of silicon area, power consumption, dynamic energy use due to variable precision arithmetic. A sample design of a 64-point FFT shows that the implementation can save about 40% area and 36% leakage power compared with a conventional parallel counterpart, accordingly achieving significant power benefits at a low sample rate and low voltage domain. The dynamic variation of the arithmetic precision can be achieved through a simple modification of the controller with hardware area overhead of 10% gate count.

Keywords: Bit-serial; Low Power; Variable Accuracy Computing; FFT; Energy Harvesting; VLSI; Hardware Design

1. Introduction

In recent years, much attention has been placed on Internet-of-Things (IoT) technologies [1][2]. A major challenge in the design of IoT sensors is the lack of reliable and continuous power sources. With advances in energy harvesting techniques, the application of self-powered sensors [3][4] has become possible [5]. Recently, a new paradigm of power-neutral computing [6] has been proposed, which remove the energy 'buffer' and instead operating directly from the harvester's variable (or transient) supply. As a result, the supply can no longer be considered to be battery-like. This pose a serious challenge, how to keep circuits alive with intermittent power supply. To cope with this constraint, the energy-harvesting-aware circuit is required to realize ultra-low-power and energy-scalable operations with unpredictable power conditions.

As for the low-power techniques, the most efficient solution is voltage scaling. NTC (Near-Threshold Computing) [7][8] has been explored where the supply voltage decreases to approximately the threshold voltage so that much of the energy savings typical of subthreshold operation can be retained with acceptable performance and robustness. Furthermore, majority of the systems powered by energy harvesting operate at low sample rate. With the near-threshold voltage and low operating frequency, the leakage power might dominate the total power consumption. In this case, bit-serial structure [9][10] is considered as a area-efficient and power-saving technique. In bit-serial architecture, all of the bits pass through the same arithmetic block in order, thereby resulting in a huge reduction in the hardware complexity. The degrader performance can be smaller as critical path in serial design is largely decreased compared with parallel counterpart. Additionally, serial implementation can realize energy-scalable operation without significant hardware cost.

As a necessary part for wireless sensor nodes, many solutions concerning various implementations of ultra-low-power FFT processors and butterfly processing elements have been developed [9][10]. As an old technique, Bit-serial operation is considered for FFT processor in the context of area-saving. In the paper, we present first study of bit-serial FFT operating at a low-voltage and exploit how much power benefits it can obtain compared with parallel counterpart with frequency scaling. The proposed

approach is analysed using 65-nm ST CMOS process, where the processing element operates at 500 mV supply.

The rest of the paper is organized as follows. Section 2 describes FFT algorithm and outlines state-of-the-art bit-serial FFT processors. Section 3 illustrates the the proposed approach to bit-serial FFT processor with variable-accuracy operation. In Section 4, we present extensive simulation results and quantify the area and power, energy savings of the proposed approach compared with standard bit-serial and parallel implementation. Section 5 summarizes and concludes the paper.

2. Background

A first bit-serial FFT implementation [11] was proposed by Wanhammar in 1995. The FFT processor uses memory-based structure and consists of four main blocks, the butterfly, the coefficient generator, the RAMs, and the control unit. In 1997, based on the precious FFT design, Wanhammar increases the number of butterfly processing elements to meet a broad spectra of throughput constraints [12]. A bit-serial FFT processor using pipeline structure was published in 2015 by Yang [13], where a trade-off between performance and power is well-balanced and area advantage against parallel counterparts is presented.

In general, the largest advantage for bit-serial FFT design is low hardware cost, while it also result in server performance penalty on the other hand. In order to meet the same throughput rate requirements with conventional parallel design, serial implementation is required to run at a faster frequency, which leads to higher power consumption. The previous publications above mainly focus on the trade-off between performance, power and area, adopting more processing units or pipeline architecture to improve circuit performance while still maintaining low-area advantage.

Nowadays, with technology scaling, the area-efficiency of bit-serial designs lead to a significant power advantage at low duty cycles or low supply voltage, where leakage is the dominant part of the power consumption. Additionally, bit-serial computing lends itself easily to variable-accuracy computing, to the best of my knowledge, it has not be demonstrated in a FFT implementation.

3. Bit-Serial FFT Butterfly with a New Bypassing Multiplier

In this design, The radix-2 butterfly datapath is adopted, which contains a complex multiplication, followed by complex-value addition or subtraction. The relevant hardware block is presented in Figure. 1, where A and B represent two complex number input data, X and Y represent two complex number output data, which would be sent back to data memory after butterfly computation and W denotes the responding FFT coefficients. For the hardware design, four signed multiplications and six signed additions or subtractions are needed. Only 10 full-adder/subtractors are used in serial's implementation. The hardware structure of the proposed bit-serial variable accuracy multiplier can be seen in Figure. 2.

As a necessary arithmetic unit in butterfly operation, a variable accuracy multiplier with row bypassing is described in the rest of section. Let two n -bit numbers to be multiplied, the multiplier Q and multiplicand M , be respectively expressed as:

$$Q = -q_{n-1}2^{n-1} + \sum_{i=0}^{n-2} q_i 2^i \quad (1)$$

$$M = -m_{n-1}2^{n-1} + \sum_{j=0}^{n-2} m_j 2^j \quad (2)$$

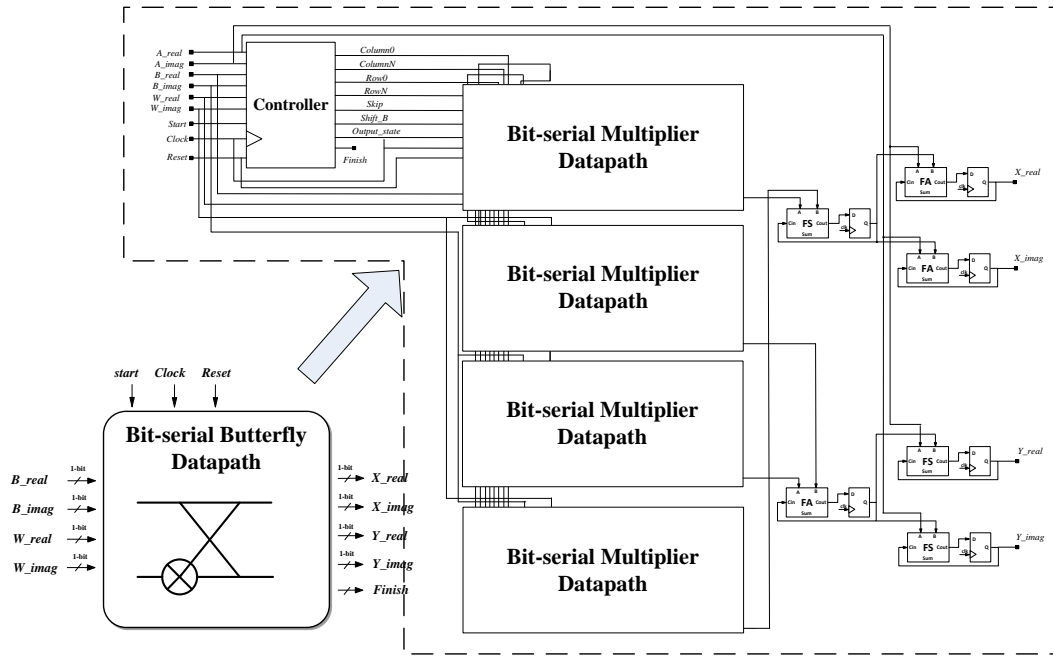


Figure 1. Block diagram of bit-serial butterfly processing element

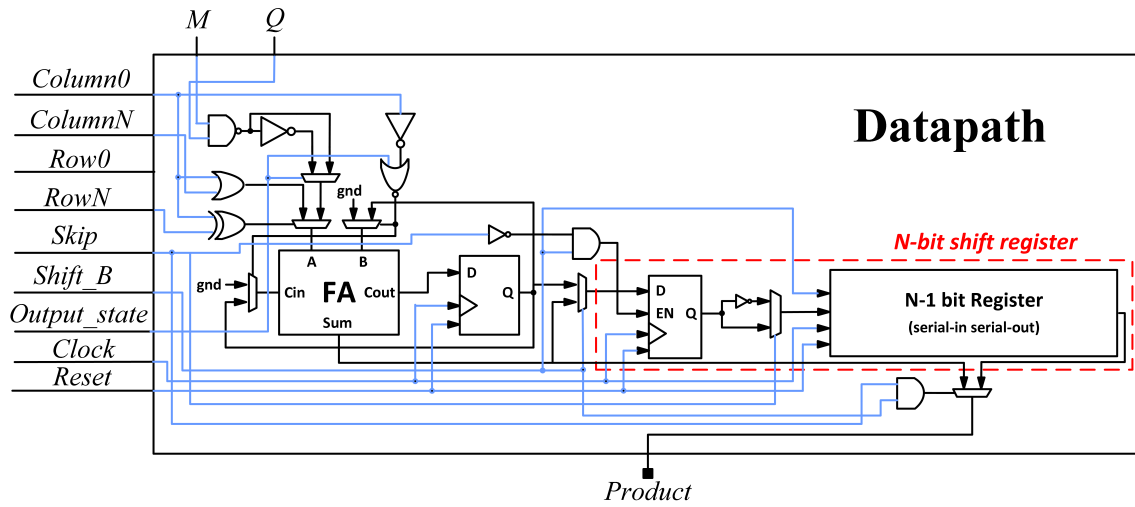


Figure 2. Structure units of the proposed bit-serial multiplier

Where q_i and m_j are the bits of Q and M correspondingly while i and j denote the bit index. Therefore, the array partial product P for bit m_j can be expressed by equation 3. While when m_j equals to zero, the partial product can be represented as $P + 2^{n-1+j}$.

$$P = \begin{cases} P + \overline{q_{n-1}}m_j2^{n-1+j} + \sum_{i=0}^{n-2} q_im_j2^{i+j}, & j < n-1 \\ P + q_{n-1}m_j2^{n-1+j} + \sum_{i=0}^{n-2} \overline{q_i}m_j2^{i+j}, & j = n-1 \end{cases} \quad (3)$$

The total operation cycle count in a conventional bit-serial multiplier is $(n+1)^2$, where the value of the product is renewed every $n+1$ cycles. We observe that, if $m_j = 0$ while index $j < n-1$, the

Algorithm 1 Calculate $P = Q \times M$ in the bypassing bit-serial way based on Baugh-Wooley algorithm

```


$$Q = -q_{n-1}2^{n-1} + \sum_{i=0}^{n-2} q_i 2^i$$


$$M = -m_{n-1}2^{n-1} + \sum_{i=0}^{n-2} m_i 2^i$$

for  $j$  from 0 to  $n$  do
  for  $i$  from 0 to  $n-1$  do
    if  $j < n-1$  and  $m_j = 0$  then
       $P = P + 2^{n-1+j}$  ▷ Row bypassing
    break
    else if  $(j < n-1 \text{ and } i < n-1) \text{ or } (j = n-1 \text{ and } i = n-1)$  then
       $P = P + q_i m_j 2^{i+j}$  ▷ Partial product accumulation
    else if  $(j < n-1 \text{ and } i = n-1) \text{ or } (j = n-1 \text{ and } i < n-1)$  then
       $P = P + \bar{q}_i m_j 2^{i+j}$  ▷ Sign complementation 1
    else if  $((j = n) \text{ and } (i = 1 \text{ or } n))$  then
       $P = P + 2^{i+j}$  ▷ Sign complementation 2
return Product  $P$ 

```

product can be accumulated by adding 2^{i+j} and, therefore, there is no need go through the entire sequence of $n+1$ cycles to update the partial product bit by bit. In other words, if there are zeros in the binary value of the multiplicand, the count of the working cycles can be reduced. The corresponding operations are presented in Algorithm 1.

4. Demonstrate on FFT Implementation

The block diagram of bit-serial FFT processor is shown in Figure. 3, using a conventional radix-2 butterfly architecture. During a butterfly computation, two complex numbers are read from data memory in serial fashion, and implement a butterfly operation with corresponding coefficients from ROMs. With assistance of scalable shift memory, computation accuracy can be adjusted in the butterfly datapath to enable adaptive-accuracy operation.

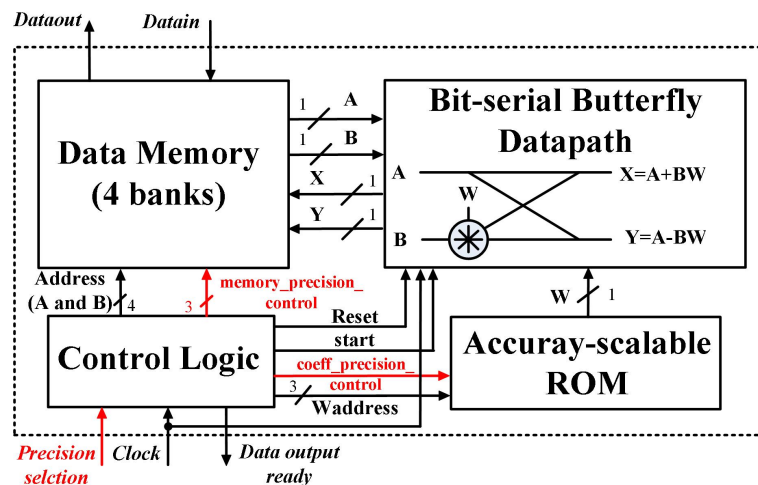


Figure 3. Structure of the proposed bit-serial FFT

Illustrated in the previous section, the butterfly datapath using proposed bit-serial multiplier, which can skip redundant clock cycles when the binary value of FFT coefficient is zero. In other words, the simple zeroing of leading or trailing bits of the coefficients, the clock cycle of butterfly operation can be decreased in proportion to the bit precision. As shown in Figure. 4(a), the twiddle factor data

output bit by bit from the ROM. The two MUXs can adjust the precision of twiddle factor value freely, one of the inputs is from the ROM output and the other is connected with ground, the two outputs would be sent to the butterfly processing block.

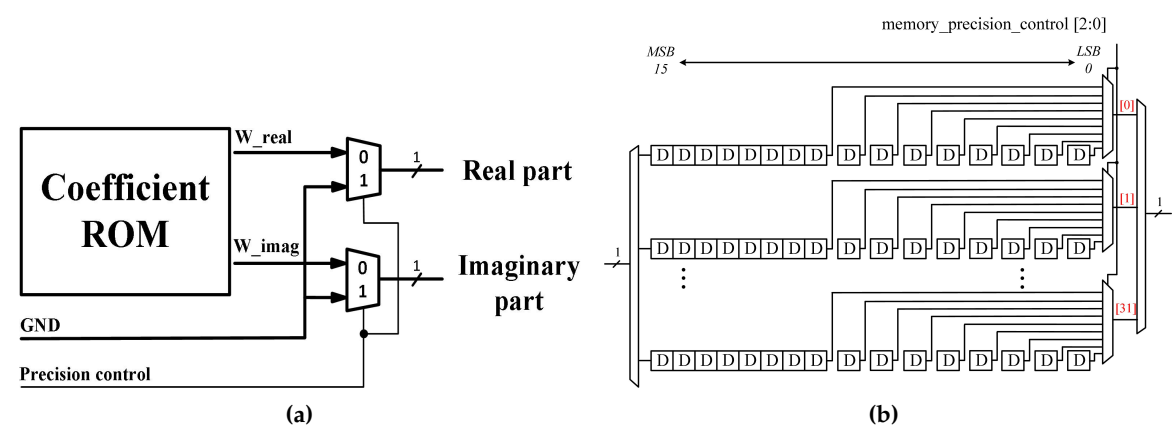


Figure 4. Block diagram of memory (a) Accuracy-scalable ROM (b) Scalable data memory based on shift registers

Additionally, the wordlength of input signals can be also reduced for further power-savings. As seen in Figure. 4(b), the proposed data memory is designed based on shift registers, the accuracy of output data is dependent on the clock cycles for shifting. A 8:1 mux is inserted following each column of shift registers to select the Least Significant Bit of the output data. Accordingly, the wordlength of input signals can be easily adjusted by the signal *memory_precision_control* from control block.

5. Case Study of a 64-Point FFT

In this section, the proposed design as well as a number of traditional FFT implementations used in our comparisons, both serial and parallel, are synthesized using the ST 65nm CMOS library and the power and energy consumption are measured by post-layout HSPICE simulations.

5.1. Scalable clock count

In general, the proposed bit-serial design supports arbitrary precision and the controller is able to perform a finer-grained selection of data bits. For simplicity, the cases of 8- and 16-bit computations are presented. As seen in Figure. 4, once 8-bit precision scaling is enabled, the data is selected from the 8th register’s output within a column of shift registers with the disability of the following 8 registers. In this way, butterfly operating cycles can be further reduced by nearly half. Specifically, in the proposed hardware design, a 16-bit FFT computation requires 26110 clock cycles, while around 43% cycles can be saved when the effective wordlength of coefficients is reduced to 8-bit. For an 8-bit operation, only 10022 cycles are needed.

Table 1. Variable wordlength of bit-serial FFT input and coefficients data with corresponding operation clock cycle count

Config	A,B Bitwdith	W Bitwdith	Operation cycles
1	16	16	26110
2	16	8	14910
3	8	8	10022

5.2. Low area and low leakage

It is well known that the serial approach uses less area and has less interconnect capacitance, which results in smaller leakage power. The 64-point FFT processors based on conventional bit-serial and energy-aware parallel architecture [16] are synthesized in our comparisons [17][18], together with other state-of-the-art serial FFT implementations. From the Table. 2, it can be found that, the proposed serial design has significantly lower hardware cost than the parallel one. The total gate numbers in the 64-point FFT design is also smaller than designs based on other serial techniques, such as digit-serial [13] and distributed arithmetic based FFT implementations [18]. It is also worth noting that compared with the conventional bit-serial design, the proposed bypassing design increases the gate count by less than 10 % . In terms of leakage power, as the design operates at a near-threshold voltage (0.5 V), additional power saving is achieved. Leakage power consumption of the serial and parallel design is 1.83 μw and 2.87 μw respectively, shown in Table. 2.

Table 2. The comparisons of gate count and leakage power between proposed bit-serial, state-of-the-art bit-serial, conventional bit-serial and parallel FFT implementations.

Architecture	Proposed	Parallel [16]	Conventional serial [11]	DA [18]	Digit-serial [13]
Gate Count	24.4K	40.5K	22K	33.7K	55K
Leakage Power	1.83 μw	2.87 μw	1.68 μw	2.44 μw	N/A

5.3. Power and energy analysis

As mentioned above, serial designs have much smaller gate counts and interconnect capacitance compared with their parallel counterparts. This advantage allows serial operations to consume significantly less power at a fixed frequency. However, as parallel implementation operate faster and require less time to complete the same operation, it can be argued that a fairer power consumption comparison should be based on a scenario where operating frequencies of parallel designs are slowed down such that all types of designs calculate their results at the same speed.

In Figure. 5, the proposed bit-serial multiplier butterfly processing unit is firstly demonstrated on the butterfly processing unit, compared with other state-of-the-art multipliers. The average power consumption of the proposed bit-serial and parallel counterparts are shown with the sample rate scaled to achieve the same computation time at both super and near-threshold voltage regions respectively. The sample rate on the x-axis is normalized to the maximum sample rate that can be processed by the conventional bit serial design, where the maximum sample rate are 4.36 MHz and 116 KHz for 1.0 V and 0.5 V voltage operations respectively. It is worth noting that both the designs are clock-gated during idle intervals. Simulation results show that proposed bit-serial design outperforms parallel counterparts at low sample rate.

The next scenario in our power consumption analysis tests the performance of the bit-serial design in FFT implementation. Both the serial and parallel FFT run at serial’s maximum sample rate, 639 KHz and 1.2 KHz for 1.0 V and 0.5 V voltage operations respectively. Simulation results in Figure. 6(a) shows that the power benefits of the proposed serial design at a given sample rate is greater at the near-threshold supply voltage. This is because the leakage power consumes a higher proportion of the overall power at low voltages. For both super and near-threshold operations, at a high sample rate, the dynamic power dominates the total power because of higher required clock frequencies. At low sample rates, it is trivial that the serial implementation’s low leakage power consumption has a significant advantage. Specifically, the results show that the proposed bit-serial design uses up to 36% less power than the parallel counterpart. In general, it turns out that the power advantage of serial butterfly datapath remains in whole FFT computation.

Figure. 6(b) displays the power comparison between the proposed serial and parallel implementations operating at 0.5 V supply voltage, with sample rate scaling, for 8-bit and 16-bit

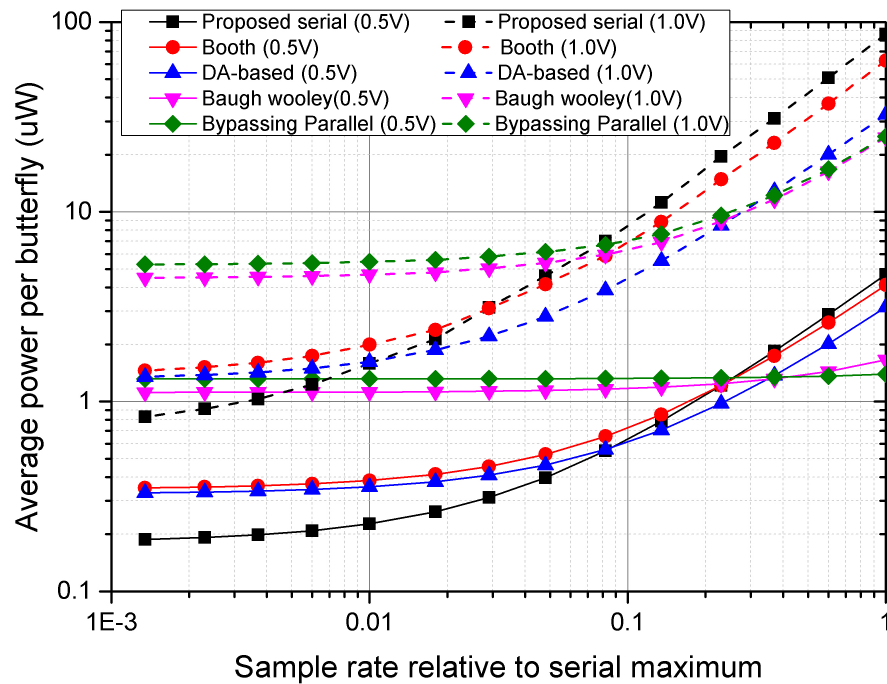


Figure 5. An average power consumption comparison in butterfly datapath with various multipliers vs sampling rate at super- and near-threshold regions (1V and 0.5V supply voltage).

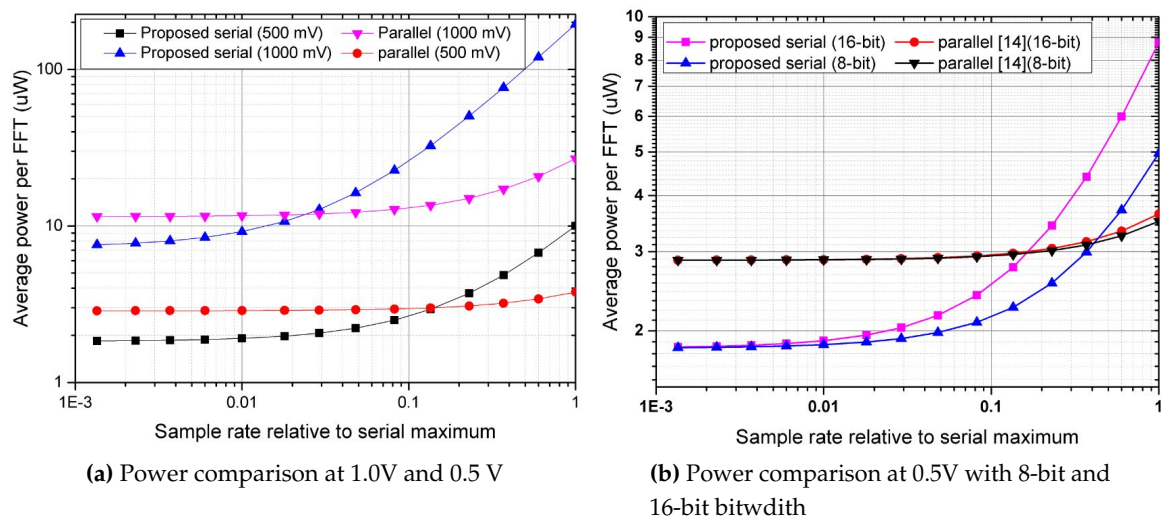


Figure 6. Average power consumption of both serial and parallel FFT implementations in terms of voltage and computation accuracy

arithmetic. It can be observed that, at low sample rate, variable-accuracy is more efficient for the proposed serial approach. For parallel implementation, although the accuracy reduction can significantly save the dynamic power in the arithmetic unit. As leakage power still domain the total power consumption due to long operation time, power saving by reduced accuracy computation in parallel FFT processor is negligible. On the other hand, serial implementation can save around 62%

clock cycles. To maintain the same computation time, operating frequency of serial design is slowed down, therefore the average power consumption can be reduced significantly.

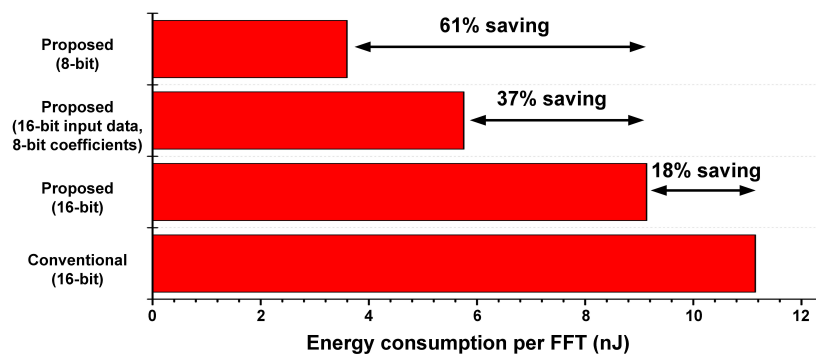


Figure 7. Energy reduction with bypssing technique and accuracy loss

In addition to the advantages of low-area and low-leakage, serial implementation lends itself easily to energy-scalable computing. Bit-serial computing can remain ultra-low power consumption during operation, while the clock cycles are reduced with the degraded computation accuracy, the energy consumption of FFT computation can also be saved. Figure. 7 presents the energy comparisons between conventional and proposed bit-serial multiplier at the minimum energy point. Also, it demonstrates the variable accuracy behaviour of the proposed design. It can be seen that, the proposed bit-serial FFT design can save around 18% energy through the bypassing technique in terms of 16-bit computation. For the variable-accuracy operation, the energy consumption changes approximately linearly with the effective bit-width as expected. Approximately 37% energy can be saved when the effective word-length of the coefficient is reduced from 16 to 8 bits while the 8-bit computation can further scales down the energy dissipation by 61%.

6. Conclusion

This paper proposes a novel bit-serial FFT processor with variable-accuracy computing which has not been investigated before. The proposed design outperforms the parallel FFT processor as well as the main types of serial implementations in terms of area and power consumption in typical application scenarios at near-threshold operation. Furthermore, by reducing the effective bit resolution of the input samples, the overall energy consumption can be scaled down dynamically. According to simulation results, when the effective wordlength is reduced from 16 to 8 bits, the total energy consumption is reduced by up to 61%. These results demonstrate that the near-threshold operation and bit-serial techniques are a promising solution to implement ultra-low power designs powered by energy harvesters, especially in IoT sensor applications where ultralow power and energy consumption are usually of greater concerns than the computational performance.

References

1. Muhic, I.; Hodzic, M. Internet of Things: Current Technological Review and New Low Power Wireless Sensor Network Protocol Proposal. *Southeast Europe Journal of Soft Computing* **2014**, *3*.
2. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems* **2013**, *29*, 1645–1660.
3. Zhang, Y.; Zhang, F.; Shakhshier, Y.; Silver, J.D.; Klinefelter, A.; Nagaraju, M.; Boley, J.; Pandey, J.; Shrivastava, A.; Carlson, E.J.; others. A batteryless 19 W MICS/ISM-band energy harvesting body sensor node SoC for ExG applications. *IEEE Journal of Solid-State Circuits* **2013**, *48*, 199–213.
4. Amirtharajah, R.; Chandrakasan, A.P. Self-powered signal processing using vibration-based power generation. *Solid-State Circuits, IEEE Journal of* **1998**, *33*, 687–695.

5. Kazmierski, T.J.; Beeby, S. *Energy harvesting systems*; Springer, New York, 2011.
6. Balsamo, D.; Das, A.; Weddell, A.S.; Brunelli, D.; Al-Hashimi, B.M.; Merrett, G.V.; Benini, L. Graceful performance modulation for power-neutral transient computing systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2016**, *35*, 738–749.
7. Dreslinski, R.G.; Wieckowski, M.; Blaauw, D.; Sylvester, D.; Mudge, T. Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits. *Proceedings of the IEEE* **2010**, *98*, 253–266.
8. Hanson, S.; Zhai, B.; Bernstein, K.; Blaauw, D.; Bryant, A.; Chang, L.; Das, K.K.; Haensch, W.; Nowak, E.J.; Sylvester, D.M. Ultralow-voltage, minimum-energy CMOS. *IBM Journal of Research and Development* **2006**, *50*, 469–490.
9. Khanna, S.; Calhoun, B.H. Serial sub-threshold circuits for ultra-low-power systems. Proceedings of the 2009 ACM/IEEE international symposium on Low power electronics and design. ACM, 2009, pp. 27–32.
10. Calhoun, B.H.; Khanna, S.; Zhang, Y.; Ryan, J.; Otis, B. System design principles combining sub-threshold circuit and architectures with energy scavenging mechanisms. Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, 2010, pp. 269–272.
11. Melander, J.; Widhe, T.; Sandberg, P.; Palmkvist, K.; Vesterbacka, M.; Wanhammar, L. Implementation of a bit-serial FFT processor with a hierarchical control structure. Proc. European Conf. on Circuit Theory and Design, ECCTD '95, Istanbul, Turkey, 1995, pp. 423–426.
12. Melander, J.; Widhe, T.; Palmkvist, K.; Vesterbacka, M.; Wanhammar, L. An FFT processor based on the SIC architecture with asynchronous PE. Circuits and Systems, 1996., IEEE 39th Midwest symposium on. IEEE, 1996, Vol. 3, pp. 1313–1316.
13. Yang, L.; Chen, T.W. A low power 64-point bit-serial FFT engine for implantable biomedical applications. Digital System Design (DSD), 2015 Euromicro Conference on. IEEE, 2015, pp. 383–389.
14. Ma, Y.; Wanhammar, L. A hardware efficient control of memory addressing for high-performance FFT processors. *IEEE transactions on signal processing* **2000**, *48*, 917–921.
15. Xiao, X.; Oruklu, E.; Saniie, J. Efficient FFT engine with reduced addressing logic. Electro/Information Technology, 2007 IEEE International Conference on. IEEE, 2007, pp. 390–395.
16. Wang, A.; Chandrakasan, A. A 180-mV subthreshold FFT processor using a minimum energy design methodology. *Solid-State Circuits, IEEE Journal of* **2005**, *40*, 310–319.
17. Yuan, B.; Wang, Y.; Wang, Z. Area-efficient scaling-free DFT/FFT design using stochastic computing. *IEEE Transactions on Circuits and Systems II: Express Briefs* **2016**, *63*, 1131–1135.
18. Jiang, M.; Yang, B.; Huang, R.; Zhang, T.; Wang, Y. Multiplierless fast Fourier transform architecture. *Electronics Letters* **2007**, *43*, 191–192.